Washington Excise Tax Microsimulation Model
2002

Rick Peterson
Office of Program Research
November 20, 2002

November 20, 2002
Rick Peterson
Office of Program Research
Washington House of Representatives

Washington Excise Tax Microsimulation Model 2002

The Washington Excise Tax Microsimulation Model is a database and set of SAS programs that can be used to assess how taxes in the current excise tax system and alternatives to that system are distributed across different classes of Washington households. The model can illustrate the tax distribution on households by income group, household size, tenure, and total outlays.

Data

The model is created from two data sets - the Washington State Population Survey and the Consumer Expenditure Survey.

The Washington State Population Survey (SPS) was designed by the Washington Office of Financial Management in a manner similar to the national Current Population Survey. The most recent SPS was conducted in the spring of 2000. The survey asked questions about employment, income, household composition (number of residents, ages, etc), and housing characteristics. The survey collected information on 6,726 Washington households.[i]

The Consumer Expenditure Survey (CEX) is a continuing survey conducted by the Bureau of Labor Statistics of the United States Department of Labor.[ii] The survey is used by the Bureau to update the structure of the consumer price index and to provide information about spending patterns of different types of families. The CEX is actually two different surveys – a diary survey and an interview survey.

The primary purpose of the diary survey is to collect expenditure information on small frequently purchased items, such as, food and beverages, housekeeping supplies, nonprescription drugs, and

personal care products. Participating households record expenditures over two week period. The diary survey covers all expenditures during each week of the survey. An interviewer from the Bureau collects demographic information from the household when retrieving the diary.

Each participant in the interview survey is interviewed every three months. Participants are asked about expenditures over the previous three months. The survey gathers data on large expenditures, such as automobiles, property, and appliances, as well as regular expenses, like rent, utility costs, and insurance premiums.

The interview survey collects very detailed information on about 60 to 70 percent of household expenditures. Additional, more aggregated, information about food and other expenditures is also obtained on 20 to 25 percent of total expenditures. So, the interview survey covers between 80 and 95 percent of total expenditures.

The SPS is used in the model to represent the distribution of Washington households across income and other classes. The CEX is used to assign spending patterns to the Washington households in the SPS data set.

Method: Combining the CEX with the SPS

In the Washington Excise Tax Microsimulation model, the SPS and CEX data sets are combined at the household/consumer unit level using a statistical matching procedure. Statistical matching is a procedure designed to provide supplemental data when a single survey does not contain all the information necessary for a desired analysis. Using statistical matching a second survey containing the desired additional information is matched to the first survey based on data that is common with the first survey. The common data is used to find and match household/consumer units that are sufficiently similar (See figure 1). [iii]

SPS households are matched with CEX consumer units based on similarities in income, tenure, housing building type, presence of person over 64 in household or consumer unit, presence of person under 18, and size. These characteristics were chosen for three reasons. First, they are common to both data sets. Second, these characteristics are likely to be used as categories when displaying the results of the analysis. Third, these characteristics are related to consumption (See appendix 1).

Figure 1
Illustration of data sets for statistical matching

	Variables in Survey A	Variables in Survey A and B	Variables in Survey B
Observations in Survey A			Missing
Observations in Survey B	Missing		

Households in SPS and the CEX

The Washington State Population Survey (SPS) contains records on households and individuals within the households. A household includes all the people that live and sleep at the residence most of the time. Each household record contains information on the number of people in the household, the household's income, the type of residence, whether the household owns or rents the residence, number of persons below age 18, whether a person in the household received social security payments and many other items.

The basic unit of analysis for the CEX is the consumer unit. A consumer unit is a family, two or more unrelated persons living together who pool their income for making expenditures, or is single person that is financially independent. For purposes of this modeling exercise households in the SPS are considered to be equivalent to consumer units in the CEX.

Household Income

Respondents to the SPS were asked to provide an exact household income. They were also asked to place their household income into one of 9 income ranges. Not all respondents provided an answer to these questions. The Office of Financial Management created a variable for total household income from the responses to these questions. The value for this variable was derived in one of three ways: it was the same as the exact income response (about 47 percent of households); it was imputed from the income range response (about 42 percent of households); or it was imputed by regression (about 11 percent of households).

The SPS also collected information on social security income, interest, dividend, and rental income, wage and other income. Negative values were not allowed for these variables.

Participants in the CEX provided detailed information on a number of income types including, wages and salaries, business income, interest, dividends, rents, pensions, unemployment and workers compensation payments, child support, public assistance, value of food stamps, and other income. Before-tax income is the sum of these items. Participants were allowed report values for negative business income and rental income.

For purposes of the microsimulation model matching procedure, before-tax income variable in the CEX data set was adjusted to remove negative values for business and rental income and to remove the value of food stamps. This was done to make the two income definitions more alike. The adjustment also corrects the problem of classifying households with high consumption patterns into low reported income categories.

Matching SPS data to CEX data

The CEX data is available at the consumer unit level in quarterly data sets – one data set for the dairy survey and one for the interview survey. A complete set of annual data is thus contained in eight data sets. The statistical matching procedure matched each household in SPS to a CEX consumer unit in each of the four diary survey data sets and the four interview data sets that make up one year of data. The matching procedure was done seven times to produce seven independent sets of matches between SPS and CEX data. See Rubin for a discussion of multiple imputation.[iv]

The statistical matching procedure selects a household from the SPS. Then potential matching candidates are selected from the CEX data based on common values of: household income, household tenure, housing building type, presence of person over 64 in household, presence of person under 18 in household, and household size.

The selection of potential candidates was done in three rounds. In the first round an exact match was required for housing tenure, building type, household size, the presence of a person under 18 in household, and the presence of a person over 64. Household income was required to be within $7,500 if SPS income was below $100,000, within $12,500 for income between $100,000 and 130,000, and within $20,000 for income over $130,000. For all households not matched in the first round, a second round of attempted matching was conducted. The second round was based on the first round parameters, but the family size was allowed to vary by one person and the income match requirement was loosened as follows: within $15,000 for SPS income below $100,000, within $25,000 for income between $100,000 and 130,000, and within $35,000 for income over $130,000. In the third round, the building type requirement was dropped for any remaining unmatched households and the family size was allowed to vary by 2 persons.

If more than one candidate satisfied the criteria for matching, a candidate was selected randomly from the group.[v]

Approximately 90 percent of the SPS households are matched in the first round with consumer units in the CEX interview survey, about 8 percent are matched in the second and about 2 percent in the third round. For matches with the CEX diary survey, about 95 percent are matched in the first round, 4 percent in the second round, and 1 percent in the third round. See Appendix 3 for details of the number of successful matches by survey and imputation round.

Outcome of Matches

To judge the performance of the matching process, tables were prepared comparing consumption in the seven matched SPS data to consumption in the original CEX data. Also, attributed consumption in the seven matched SPS data sets were compared across the seven matched SPS datasets. These tables were done for consumption reported in the CEX interview survey.

Table 1 compares the average consumption by income category[vi] for the CEX data and the average of the seven SPS matched data sets. Appendix 4 contains charts comparing average consumption for the CEX data and the SPS matched data for selected detail expenditure categories broken out by income group. These tables and charts show a close correspondence between consumption reported in the original data and consumption in the matched SPS data sets.

Appendix 5 shows the average consumption for the seven SPS matched data sets for selected expenditure categories and also shows the range of results for the seven SPS matched data sets. The charts show very similar consumptions levels for most expenditure categories in each of the seven data sets. The range is wider for vehicle purchases than for other expenditure categories. This is expected because not all households purchase a vehicle each year.

Table 1

Average Consumption by Income Category

Average for CEX data compared to Average for the 7 Imputation Groups

	Total Expenditures	
	CEX	SPS Matched
	Average	Average
1999 HOUSEHOLD TOTAL INCOME		
$0 to $20,000	18,416	19,120
$20,000 to $30,000	26,955	25,410
$30,000 to $40,000	32,304	31,245
$40,000 to $50,000	38,451	37,874
$50,000 to $60,000	44,077	44,054
$60,000 to $70,000	49,023	49,470
$70,000 to $80,000	56,770	54,284
$80,000 to $100,000	60,581	62,017
$100,000 to $130,000	74,764	72,966
Over $130,000	105,952	99,786

Calibrating the Model

The model estimates the distribution of the sales and use tax and a number of the special excise taxes across households. The taxes are calculated by multiplying expenditures on items subject to the taxes by the tax rates. This section describes the expenditure items subject to each of the taxes, the adjustment to consumption expenditures to reflect under reporting in the CEX, and a comparison of aggregate tax estimates to independent estimates of the amount of these taxes paid by households.

Identifying the tax bases

The Consumer Expenditure Survey collects data on over 900 separate CEX expenditure and income categories. Appendix 6 is a table showing the categories in the survey subject to Washington sales and use tax. Expenditure categories subject to retail sales tax are identified in the column labeled "Current". Items that are fully taxable have a value of "1". For items that are partially taxable, e.g. home telephone services – where the basic residential service is exempt but other telephone services are not, the value is a fraction.[vii]

Expenditure items subject to other excise taxes are coded in the column labeled "Other". The codes are used to calculate the impact of these other excise taxes. The codes are as follows:

Table 2

Tax	Code
Beer (volume tax sold in original container)	1
Wine (volume tax on wine sold in original container)	2
Liquor (volume tax on liquor sold in original container)	3
Beer (volume tax sold by the drink)	4
Wine (volume tax on wine sold by drink)	5
Liquor (special sales tax on liquor sold by drink)	6
Insurance Premiums Tax (gross receipts)	7
Cigarette Tax (volume tax)	8
Other Tobacco Products Tax (wholesale value tax)	9
Public Utility Tax on Electricity	11
Public Utility Tax on Natural Gas	12
Public Utility Tax on Water/Sewer	13
Public Utility Tax on Garbage Collection	14
Gas Tax (volume tax)	15
Public Utility Tax on intercity transportation	17
Public Utility Tax on intracity transportation	18

Adjusting the data

To account for discrepancies between reported consumption levels in some categories and actual levels implied by tax collections, the amount of consumption reported in the CEX was adjusted. These discrepancies exist, for example, in the reported consumption for items such as alcoholic beverages and tobacco products. In addition, other expenditure categories are also underreported.[viii] Based on BLS publication that compares reported survey expenditures with independent estimates , the amount of spending was adjusted. In addition, some further adjustments were made so that aggregate tax revenue from households match estimates of revenue for the specific revenue sources such as alcohol taxes, tobacco taxes, and the gasoline tax.

Comparing aggregate tax estimates with model results

The Washington Department of Revenue estimates that 60 percent of the retail sales and use tax is paid directly by households. It would therefore be expected that for calendar year 1999 sales

taxable consumption within the excise tax microsimulation model would yield a state sales and use tax of $3.2 billion. Table 3 shows the estimated revenue collection from households for the state sales and use taxes and other excise taxes compared to the amount of tax estimated by the excise tax microsimulation model for the average of the seven imputation groups. The results for each imputation group are found in Appendix 7.

Table 3

Tax	Estimated Household Tax Collections (1999)	Model Estimate (Average of 7 imputation groups)
State Sales and Use Tax	$3.319 billion	$3.395 billion
Beer Tax (sold in original contained and by drink)	$29.6 million	$29.7 million
Wine Tax (sold in original container and by drink)	$12.9 million	$12.9 million
Liquor Volume Tax (sold in original container)	$46.0 Million	$43.0 million
Liquor Sales Tax (sold in original container)	$40.5 million	$40.0 million
Liquor Volume Tax (sold by the drink)	$16.9 million	$16.4 million
Liquor Sales Tax (sold by the drink)	$9.3 million	$9.3 million
Cigarette Tax	$328 million	$330 million
Other Tobacco Products	$32.9 million	$32.9 million
Public Utility Tax on Electricity	$64.0 million	$63.8 million
Public Utility Tax on Natural Gas	$13.6 million	13.6 million
Public Utility Tax on Water/Sewer	$13.9 million	$14.0 million
Public Utility Tax on Garbage Collection	$10.0 million	$10.0 million
Gasoline Tax	$508 million	$507 million
Public Utility Tax on inter and intra city transportation	$7.7 million	$7.9 million

RESULTS

Excise Taxes by Income Category

Table 5[ix] shows the average excise taxes paid by income group. The income definition is total household income from the SPS. Table 6 shows percent of income paid in tax by income category. The table includes all the state excise taxes listed above and local sales taxes and local taxes on utilities. The percent of income paid in tax declines as income rises. See Appendix 8 for graphs of the total excise tax by income group and total excise tax as a percent of income. These graphs also show the interquartile range (the households between the 25th and 75th percentile. Tax as a percent of income is lower here than found in other estimates.[x] This is explained, in part by the reclassification of some high spending households from the lowest

income categories to higher income categories by removing business losses from the definition of income. In addition, other studies often allocate taxes paid directly by businesses to households.

Table 5

State and Local Excise Tax by Income Category

Average of all Imputation Groups

	Total Excise Taxes	Sales Tax	Alcohol Taxes	Ins Tax	Tobacco Taxes	Utility Taxes	Gas tax
	Sum	Sum	Sum	Sum	Sum	Sum	Sum
1999 HOUSEHOLD TOTAL INCOME							
$0 to $20,000	1,158	785	33	28	135	65	111
$20,000 to $30,000	1,551	1,084	38	43	147	80	159
$30,000 to $40,000	1,901	1,355	56	53	153	84	199
$40,000 to $50,000	2,284	1,647	58	63	182	98	236
$50,000 to $60,000	2,663	1,975	75	68	190	106	249
$60,000 to $70,000	2,947	2,208	75	75	193	114	281
$70,000 to $80,000	3,221	2,454	89	84	188	119	286
$80,000 to $100,000	3,579	2,780	99	89	181	122	308
$100,000 to $130,000	3,998	3,244	93	95	129	134	303
Over $130,000	5,507	4,673	129	102	138	163	302

Table 6

State and Local Excise Tax as Percent of Income

Average of all Imputation Groups

	Total Excise Taxes	Sales Tax	Alcohol Taxes	Ins Tax	Tobacco Taxes	Utility Taxes	Gas Tax
	Sum	Sum	Sum	Sum	Sum	Sum	Sum
1999 HOUSEHOLD TOTAL INCOME							
$0 to $20,000	9.90%	6.72%	0.28%	0.24%	1.15%	0.56%	0.95%
$20,000 to $30,000	6.34%	4.43%	0.15%	0.17%	0.60%	0.33%	0.65%
$30,000 to $40,000	5.57%	3.97%	0.17%	0.15%	0.45%	0.25%	0.58%
$40,000 to $50,000	5.15%	3.71%	0.13%	0.14%	0.41%	0.22%	0.53%
$50,000 to $60,000	4.95%	3.67%	0.14%	0.13%	0.35%	0.20%	0.46%
$60,000 to $70,000	4.60%	3.45%	0.12%	0.12%	0.30%	0.18%	0.44%
$70,000 to $80,000	4.35%	3.31%	0.12%	0.11%	0.25%	0.16%	0.39%
$80,000 to $100,000	4.07%	3.16%	0.11%	0.10%	0.21%	0.14%	0.35%
$100,000 to $130,000	3.57%	2.90%	0.08%	0.09%	0.12%	0.12%	0.27%
Over $130,000	2.75%	2.33%	0.06%	0.05%	0.07%	0.08%	0.15%

Taxes by Total Outlay Categories

The purpose of classifying households and observing their tax burden by the selected classification is to evaluate the fairness of the tax system. The preceding section used current income as the classification method under the assumption that income represents a measure of the ability to pay. An alternative way to classify households is by total household consumption. This can be justified in two ways. First, a household's consumption is a measure of the use the household makes of the economy's resources. Second, consumption can be viewed as a proxy for permanent income under the permanent income hypotheses.[xi] The underlying idea behind the permanent income hypothesis is that consumption is made up of permanent consumption (equal to the household's permanent income where permanent income is defined as an annual income stream with a present value equal to the household's wealth) and transitory consumption. For each household, total consumption may be above or below permanent consumption depending

on whether transitory consumption is positive or negative. When summed across households, it is assumed that transitory consumption is zero.

The definition of total outlays used in this model is a modification of the CEX definition of total expenditures (which includes costs of goods and services, excise and sales taxes, purchases of financed vehicles, home mortgage interest payments, personal insurance, contributions to retirement and pensions, gifts, and contributions). Outlays includes total expenditures with the following modifications: purchase price of financed vehicles is excluded and the principal payment for financed vehicles is included, the principal payment on home mortgages is included, and contributions for retirement and pensions are excluded.[xii] See Appendix 9 for a table showing the relationship between total income and total outlays.

Table 7 shows the state and local excises tax paid by households by total outlay category. As expected the amount of tax rises with outlays.

Table 7

State and Local Excise Tax by Total Outlay Category

Average of all Imputation Groups

	Total Excise Taxes	Sales Tax	Alcohol Taxes	Ins Tax	Tobacco Taxes	Utility Taxes	Gas Tax
	Sum	Sum	Sum	Sum	Sum	Sum	Sum
Total Outlays $0 to $15,000	811	512	33	21	109	54	82
$15,000 to $20,000	1,247	831	43	34	133	70	136
$20,000 to $25,000	1,610	1,105	52	46	151	81	176
$25,000 to $30,000	2,056	1,462	63	56	168	92	215
$30,000 to $35,000	2,420	1,752	69	65	187	103	243
$35,000 to $40,000	2,784	2,069	73	71	193	110	267
$40,000 to $45,000	3,077	2,331	81	82	184	120	280
$45,000 to $55,000	3,526	2,757	84	89	170	127	299
$55,000 to $70,000	4,377	3,556	102	97	167	139	317

	Total Excise Taxes	Sales Tax	Alcohol Taxes	Ins Tax	Tobacco Taxes	Utility Taxes	Gas Tax
	Sum	Sum	Sum	Sum	Sum	Sum	Sum
Over $70,000	6,553	5,685	108	118	151	166	325

Table 8 shows state and local excise taxes as a percent of total outlays. For alcohol, tobacco, and utility taxes the percent of outlays paid in tax is higher for households with lower total outlays. Tax as a share of total outlays is nearly equal across households for insurance and gas taxes. The share of outlays paid in sales tax is higher for higher spending households. Over all, excise taxes as a percent of total outlays is about the same across households when households are arrayed by spending. See Appendix 10 for graphs of excise taxes by outlay category. These graphs include the interquartile range for each outlay group.

Table 8

State and Local Excise Tax as Percent of Total Outlays

Average of all Imputation Groups

	Total Excise Taxes	Sales Tax	Alcohol Taxes	Ins Tax	Tobacco Taxes	Utility Taxes	Gas Tax
	Sum	Sum	Sum	Sum	Sum	Sum	Sum
Total Outlays							
$0 to $15,000	6.91%	4.37%	0.28%	0.18%	0.92%	0.46%	0.70%
$15,000 to $20,000	7.08%	4.72%	0.24%	0.19%	0.75%	0.40%	0.77%
$20,000 to $25,000	7.15%	4.90%	0.23%	0.20%	0.67%	0.36%	0.78%
$25,000 to $30,000	7.48%	5.32%	0.23%	0.20%	0.61%	0.33%	0.78%
$30,000 to $35,000	7.46%	5.40%	0.21%	0.20%	0.58%	0.32%	0.75%
$35,000 to $40,000	7.44%	5.53%	0.20%	0.19%	0.52%	0.29%	0.71%
$40,000 to $45,000	7.25%	5.49%	0.19%	0.19%	0.43%	0.28%	0.66%

	Total Excise Taxes	Sales Tax	Alcohol Taxes	Ins Tax	Tobacco Taxes	Utility Taxes	Gas Tax
	Sum	Sum	Sum	Sum	Sum	Sum	Sum
$45,000 to $55,000	7.13%	5.57%	0.17%	0.18%	0.34%	0.26%	0.61%
$55,000 to $70,000	7.12%	5.78%	0.17%	0.16%	0.27%	0.23%	0.51%
Over $70,000	6.73%	5.84%	0.11%	0.12%	0.15%	0.17%	0.33%

Excise and Property Taxes paid by Households

Estimates of property taxes paid by homeowners and renters are incorporated in the following tables. The homeowner property tax estimates come from the Homeowner Property Tax Model.[xiii] Renters are assume to pay property tax as part of their rents. Residential renter property taxes are estimated by multiplying the monthly rent reported in the SPS by estimates of gross rent multipliers obtained from county assessor offices. In urban areas it was assumed that single family residences were valued at 140 times monthly rent and multi-family residences at 96 times monthly rent. In rural areas it was assumed that single family residences were valued at 120 times monthly rent and multi-family residences at 80 times monthly rent. Average property tax rates by region were used to calculate the property taxes.

Table 9

State and Local Excise and Property Tax as Percent of Income
Average of all Imputation Groups

	Total Excise and Prop Taxes	Total Excise Taxes	Prop Tax
	Sum	Sum	Sum
1999 HOUSEHOLD TOTAL INCOME			
$0 to $20,000	1,838	1,158	680
$20,000 to $30,000	2,402	1,551	851
$30,000 to $40,000	3,218	1,901	1,317
$40,000 to $50,000	3,706	2,284	1,422
$50,000 to $60,000	4,393	2,663	1,730

	Total Excise and Prop Taxes	Total Excise Taxes	Prop Tax
	Sum	Sum	Sum
$60,000 to $70,000	4,937	2,947	1,990
$70,000 to $80,000	5,479	3,221	2,257
$80,000 to $100,000	5,947	3,579	2,368
$100,000 to $130,000	6,770	3,998	2,771
Over $130,000	9,277	5,507	3,771

Table 10

State and Local Excise and Property Tax as Percent of Income

Average of all Imputation Groups

	Total Excise and Prop Taxes	Total Excise Taxes	Prop Tax
	Sum	Sum	Sum
1999 HOUSEHOLD TOTAL INCOME			
$0 to $20,000	15.67%	9.90%	5.77%
$20,000 to $30,000	9.82%	6.34%	3.48%
$30,000 to $40,000	9.43%	5.57%	3.86%
$40,000 to $50,000	8.35%	5.15%	3.21%
$50,000 to $60,000	8.17%	4.95%	3.21%
$60,000 to $70,000	7.71%	4.60%	3.11%
$70,000 to $80,000	7.40%	4.35%	3.05%
$80,000 to $100,000	6.76%	4.07%	2.69%
$100,000 to $130,000	6.04%	3.57%	2.47%
Over $130,000	4.48%	2.75%	1.73%

Using the Microsimulation Model to Calculate the Impact of Alternatives

In this section two types of tax alternatives will be discussed: changes to the tax base and changes in tax rates.

Tax Base Changes

Tax base changes are made by specifying a different set of consumption items to apply to the tax. This is done on the spreadsheet called 'Taxable Items List.xls'. This spreadsheet is a table of 913 consumption items available in the Diary and Interview Surveys. An item is included in the tax base by placing a '1' in the appropriate column and is excluded by placing a '0'; The column labeled 'Current' is reserved for the current sales tax base. Two columns are provided for alternative tax base choices. The column labeled 'Other' contains the codes for the miscellaneous excise taxes. 'Baseadj' is used to calibrate the model. (see the section above on Calibrating the Model).

The SAS program called 'Create Combined Consumption Tax Base' (see Appendix 13) reads in the CEX data and the selections in 'Taxable Items List.xls'. The program uses this information to create the current and alternative tax bases. The program uses the matches between the CEX and the SPS households to create a record for each SPS household with information on the tax bases for the current sales tax, alternative sales taxes, and the miscellaneous taxes.

Short outline of the 'Create Combined Consumption Tax Base' program

1. Read Diary and Interview Data files (These data sets are quarterly so 8 data sets are read in.)
2. Combine the data sets from step 1 into one data set. (A permanent data set is created so that step 1 can be skipped once it has been created the first time.)
3. Read in the Taxable Items List.
4. Merge consumption and taxable items and calculate tax bases for excise taxes.
5. Separate data into quarterly tax base data sets by Diary and Interview. This generates (2 data sets by 4 quarters by 7 imputation groups) 56 data sets. The observations in each data set are summed by CEX household identification number.
6. Bring in the files that match CEX households with SPS households and assign the appropriate SPS households to each observation in the tax base data sets.
7. Combine the Diary data sets and annualize the weekly expenditures.
8. Combine the Interview data sets.
9. Combine the Diary and Interview data sets and sum by SPS household. This creates one data set for each of the seven imputation groups. Permanent data sets are created so that tax rate changes can be run without recreating the tax bases.

Tax Rate Changes

The tax amounts are generated by the program called 'Tax Calculator for Microsimulation Model'. (See appendix 13) The program calculates the tax from the current tax rates and alternative tax rates. Tables are generated showing the total tax amounts and the distribution of taxes across income categories. The program brings in the tax bases created by 'Create Combined Consumption Tax Base'. The program calculates the tax for each of the seven imputation groups and averages the results.

Tax Calculator for SimTax Model

The Excise Tax Microsimulation Model was used to generate output for use in the SimTax[xiv] Model. The SimTax model was developed at the request of the Washington Tax Structure Study Committee, and was used by committee members in developing their tax alternative proposals. SimTax is a Microsoft Excel spreadsheet that allows the user to change elements of the Washington State tax system and see the impact on revenue and the distribution of tax burdens on households. The household tax distribution contained in SimTax for alternatives related to sales tax, property tax, individual income tax , value added tax, and goods and services tax was derived using the Excise Tax Microsimulation Model.

Appendix 14 contains the program called 'Tax Calculator for SimTax Model'. It is structured similar to 'Tax Calculator for Microsimulation Model'. Additional programming steps have been added to simulate aspects of a personal income tax. One personal income tax considered by the Washington Tax Structure Study Committee included a standard deduction of $5,000 for single returns and for married households $7,000 plus the smaller of $3,000 or the earnings of the second wage earner. The program called 'Second Earner Wages for Married Households' (also in Appendix 14) establishes the amount of the second earner wages from the SPS data. The simulation shown here illustrates the impact of a graduated income tax using this standard deduction (no itemizing allowed) and rates of 2% on first $49,900 of income, 3% on income between $49,900 and $120,650, and 5% on income above $120,650. (Single return brackets are half these amounts.)

The Committee also considered a goods and services tax. The goods and services tax is a sales tax which covers all goods and services. The impact of this tax was simulated by including in the alternative tax base all the goods and services available in the 'Taxable Items List.xls'.

The results of the simulation are shown in Table 11.

Table 11

Average of all Imputation Groups

	State and Local Sales Tax	Goods and Services Tax (1% rate)	Grad Rate Inc Tax (2%, 3%, 5% rates)
	Sum	Sum	Sum
1999 HOUSEHOLD TOTAL INCOME			
$0 to $20,000	785	169	46
$20,000 to $30,000	1,084	226	231
$30,000 to $40,000	1,355	272	412
$40,000 to $50,000	1,641	327	610

	State and Local Sales Tax	Goods and Services Tax (1% rate)	Grad Rate Inc Tax (2%, 3%, 5% rates)
	Sum	Sum	Sum
$50,000 to $60,000	1,975	371	814
$60,000 to $70,000	2,208	419	1,018
$70,000 to $80,000	2,454	460	1,281
$80,000 to $100,000	2,780	515	1,709
$100,000 to $130,000	3,244	589	2,552
Over $130,000	4,593	839	7,379

[i] See the Office of Financial Management's web site for detailed information on the SPS: http://www.ofm.wa.gov/sps/2000/index.htm. The data set used in this model is the Oct 1, 2001 release (sps00_04).

[ii] BLS Handbook of Methods, 1997, Chapter 16. Consumer Expenditures and Income

[iii] Statistical matching is a common technique for the construction of microsimulation models. See Cohen for a survey of statistical matching in micosimulation models. Statistical matching is also used in the fields of media research and data mining where it is often called data fusion. Cohen M, L (1991), 'Statistical Matching and Microsimulation Models', in C F Citro and E A Hanushek (eds), Improving Information for Social Policy Decisions. The Uses of Microsimulation Modeling vol. II National Academy Press, Washington D.C.

[iv] See Rubin, D.B. (1987). Multiple Imputation for Nonresponse in Surveys. New York:Wiley. and Little, R.J.A and Rubin, D.B. (1987). Statistical Analysis with Missing Data. New York: Wiley.

[v] Since the CEX is a sample, each observation has a weight associated with it. The likelihood of selection of a particular candidate was proportional to the relative weight of that candidate within the selected group. See Appendix 2 for the SAS programs that select the matches.

[vi] The income definition used for the CEX data is the before tax income (from the CEX) adjusted by removing negative values for business and rental income and the value of food stamps. The income definition used for the SPS column is SPS defined total household income.

[vii] Also, some hotel/motel and car rentals on out-of-town trips are assumed to take place outside the state and so are exempt from Washington's tax. The expenditure amount reported for vehicles is the amount spent net of any trade-in. Under Washington's sales tax the trade-in value of vehicles is offset against the price of new and used vehicles. So, the net outlay number is used directly and the value of the trade-in is not included in the sales tax base.

[viii] As part of their postsurvey evaluations, the Bureau of Labor Statistics compares the expenditures reported in the survey with other independent sources of household expenditures. The most recent comparison was published in 1994. See Branch, E. Raphael, 'The Consumer Expenditure Survey: a comparative analysis', Monthly Labor Review Dec 1994.

[ix] Table 4 is intentionally left out.

[x] See Citizen's for Tax Justice and Institute on Taxation and Economic Policy, Who Pays? A Distributional Analysis of the Tax Systems in All 50 States, http://www.ctj.org/htm/whopay.htm

[xi] See Friedman, M. (1957) A Theory of the Consumption Function. Princeton, Princetion University Press, and the extensive literature that followed.

[xii] See Rogers, John M, and Maureen B. Gray, "CE data: quintiles of income versus quintiles of outlays", Monthly Labor Review, Dec 1994. They, however, include pension and retirement contributions in their definition of total outlays.

[xiii] For documentation see: Homeowner Property Tax Model 2002, Rick Peterson, Office of Program Research, Washington House of Representatives, October 22, 2001.

[xiv] SimTax is available on the Washington Dept of Revenue web site see: http://dor.wa.gov/content/WAtaxstudy/Tax_Design.htm

Regression on Total Expenditures

The REG Procedure

Model: MODEL1

Dependent Variable: totexppq Total Expenditures

Weight: normwgt

Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	6	2.485081E11	41418022279	1192.22	<.0001
Error	22779	7.913452E11	34740119		
Corrected Total	22785	1.039853E12			

Root MSE	5894.07489	R-Square	0.2390
Dependent Mean	5813.60123	Adj R-Sq	0.2388
Coeff Var	101.38423		

Parameter Estimates						
Variable	Label	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	Intercept	1	1549.37207	102.22778	15.16	<.0001
fincbtax	Income before taxes	1	0.25978	0.00379	68.58	<.0001
Fam_size	Household size	1	319.79515	45.70479	7.00	<.0001
under18	Person under 18 in household	1	87.58925	131.77759	0.66	0.5063
over64	Person over 64 in household	1	-534.02303	102.66003	-5.20	<.0001
own	Homeowner	1	579.68900	99.51669	5.83	<.0001

Parameter Estimates						
Variable	**Label**	**DF**	**Parameter Estimate**	**Standard Error**	**t Value**	**Pr > |t|**
bld	Single family detached residence	1	570.30161	96.85038	5.89	<.0001

```
*************************************************************************;
* program:    Hot decking Diary CEX to WAPOP survey multiple imp         ;
*             with donor wgt adj.sas[i]                                   ;
* programmer: Rick Peterson                                              ;
* project:    Washington Excise Tax Microsimulation Model                ;
* date:       March 21, 2002                                             ;
*                                                                        ;
* purpose:    Match households in WA pop survey with similar households  ;
*             in Consumer Expenditure Survey. Matching variables: Income,;
*             housing tenure, housing building type, presence of person  ;
*             over 64 in household, presence of person under 18 in       ;
*             household, and family size.                                ;
*                                                                        ;
*------------------------------------------------------------------------;
* libraries: popsur - location of WAPOP survey data                      ;
*            mdimpute - destination of matching files                    ;
*                                                                        ;
*------------------------------------------------------------------------;
* incoming:   From Diary CEX: fmly99.1, fmly99.2, fmly99.3 fmly99.4       ;
*             From WA population survey: sps00f04                        ;
*                                                                        ;
*------------------------------------------------------------------------;
* formats:    Format for WAPOP survey                                     ;
*                                                                        ;
*------------------------------------------------------------------------;
* outgoing:   DImpute1, DImpute2, DImpute3, DImpute4: containing household;
*             matches.                                                   ;
*             dqtrsmatched: containing number of quarters matched for    ;
*             each household - used in later program to adjust consumption;
*             data.                                                      ;
*                                                                        ;
*------------------------------------------------------------------------;
* reports:    None                                                       ;
*                                                                        ;
*------------------------------------------------------------------------;
* changes:                                                               ;
*                                                                        ;
*------------------------------------------------------------------------;
* notes:      Each Washington household is matched to a CEX household for ;
*             each of the 4 quarter of CEX data.  Matching is done in    ;
*             three rounds.  If a match is not found then matching       ;
*             criteria are relaxed for next round.  When more than one   ;
*             donor is available then the donor is selected randomly.    ;
*             The likelihood of donor selection is proportional to the   ;
*             relative weight of the donor within the group of potential ;
*             donors.                                                    ;
*                                                                        ;
*                                                                        ;
*************************************************************************;

*------------------------------------------------------------------------;
*Read in donor pool from CEX diary data sets;
*------------------------------------------------------------------------;

%let y = 99;
filename dfmly1 "c:\data\diary survey\fmlyd&y.1.txt";
filename dfmly2 "c:\data\diary survey\fmlyd&y.2.txt";
```

```
filename dfmly3 "c:\data\diary survey\fmlyd&y.3.txt";
filename dfmly4 "c:\data\diary survey\fmlyd&y.4.txt";

%macro loop1;
%do x= 1 %to 4 %by 1;
data dfmly&x;
   infile dfmly&x lrecl=1549;
   input @1 newid 8.
            @148 finlwt21 11.3
            @43 cutenure $1.
            @78 Fam_size 2.
         @544 perslt18 2.
            @582 respstat $1.
            @36 age_ref 2.
            @547 perot64 2.
            @139 fincbtax 8.
            @45 Descrip $2.
            @83 fbsnsx 8.
            @103 ffarmx 8.
            @584 roomx 8.
            @526 othrntx 8.
            @423 jfs_amt 8.
            ;
*Eliminate obs without complete reporting of income;
if respstat ne '1' then delete;
*Create a variable to merge with WAPOP survey observation;
Flag = 1;
*Create variable to match housing tenure;
if cutenure in ('1', '2','3') then own = 1;
else if cutenure eq '4' then own =2;
else if cutenure eq '5' then own =3;
else own =4;
*Create variable to match building type;
if Descrip in ('01') then bld = 1;
else if Descrip in ('02','03','04') then bld = 5;
else if Descrip in ('05', '06') then bld = 3;
else bld = 5;
*Cap household size at 6 people;
if fam_size >5 then fam_size = 6;
*Create variable to match households with persons under 18;
if perslt18>0 then under18 = 1;
   else under18=0;
*Create variable to match households with persons over age 64;
if perot64>0 then over64 = 1;
   else over64 = 0;
*Remove business losses and value of food stamps from income;
   if fbsnsx ne . and fbsnsx <0 then fincbtax = fincbtax - fbsnsx;
   if ffarmx ne . and ffarmx <0 then fincbtax = fincbtax - ffarmx;
   if roomx ne . and roomx <0 then fincbtax = fincbtax - roomx;
   if othrntx ne . and othrntx <0 then fincbtax = fincbtax - othrntx;
   if jfs_amt ne . then fincbtax = fincbtax - jfs_amt;
*Drop household that is in fmly data set but has no consumption information
   in expn data set;
   %if &x=1 %then %do;
    if newid=189172 then delete;
    if newid=189171 then delete;
   %end;
```

```
   %if &x=2 %then %do;
    if newid=204201 then delete;
    if newid=204202 then delete;
   %end;
run;
%end;
%mend Loop1;
%Loop1;

*------------------------------------------------------------------------;
*Read in data from WAPOP survey;
*------------------------------------------------------------------------;

Data one;
set popsur.sps00f04;
*Create variable for presence of person over64 in household;
if age >64 then over64popsur =1;
else over64popsur =0;
keep id age over64popsur;
run;

proc summary data=one;
by id;
var over64popsur;
output out=two sum=;
run;

Data A;
set popsur.sps00f04;
where pnum=1;
Flag = 1;
*Cap household size at 6 people;
if peopl >5 then peopl = 6;
*Create variable to match persons in household under 18;
if chldrn18>0 then WAsurvey18 = 1;
   else WAsurvey18=0;
*Change building type variable to match CEX response possibilities;
   if q3p1 in (1,2) then bldpopsurvey = 1;
   else if q3p1 = 3 then bldpopsurvey = 3;
   else bldpopsurvey = 5;
keep Flag ID pnum age bldpopsurvey q3p2 fnlwgt hhinc peopl WAsurvey18;
run;

*Create "Missing" - the  WAPOP data set with household information but
without consumption data;
*Merge in variable for presence of person over 64;
Data missing;
merge A two;
by id;
if over64popsur >0 then over64popsur=1;
else over64popsur=0;
drop _type_ _freq_;
run;

*------------------------------------------------------------------------;
*FIRST ROUND OF IMPUTATON - Restrict potential donors by tenure,
property type, family size, age and income;
```

```
*------------------------------------------------------------------------;
*
Turn off writing to the SAS log;
options nonotes nosource nosource2 errors=0;

*Create macro loop to find a match for the WAPOP household in each
 of the 4 CEX diary data sets;

%macro loop;
%do x = 1 %to 4;

*Count the number of obs in missing;

proc means data = missing noprint;
var flag;
output out=amiss sum=;
run;

*Delete file which will contain imputed values;

Proc delete data=mdimpute.ADImpute&x;
run;

*Create file which will contain imputed values;

Data mdimpute.ADImpute&x;
set missing;
keep ID;
Proc sort;
by ID;
run;

*Create macro variable equal to number of obs with missing CEX data;

Data num;
set amiss;
call symput('num',left(_freq_));

*Macro function to do imputation;

%macro Impute(a=);

*Loop which starts with first obs with missing data and ends with last one;

%do %until (&a gt &num);

*Pull the next obs with missing data;

Data two;
Set missing;
If _N_ =&a;

*Merge Donor data set with missing data set;
*Keep only Obs from Donor data set with similar characteristics;

Data threea;
Merge Dfmly&x Two; By Flag;
```

```
if own ne q3p2 and q3p2 in (1,2,3)
then delete;                                *Requires exact match for tenure;
if bld ne bldpopsurvey and bldpopsurvey in (1,3,5)
then delete;                                *Requires exact match for bld type;

if over64popsur ne over64 then delete; *Requires presence of person over64;

if hhinc <100000 then do;
   if fincbtax gt (hhinc +7500)
   or fincbtax lt (hhinc-7500) then delete; *Requires income to be within
7,500 if income below 100,000;
   end;
if 100000<= hhinc <= 130000 then do;
      if fincbtax gt (hhinc+12500)
      or fincbtax lt (hhinc-12500) then delete;*Requires income to be within
12,500 if income above 100,000 and below 130,000;
      end;
if hhinc > 130000 then do;
      if fincbtax gt (hhinc+20000)
      or fincbtax lt (hhinc-20000) then delete;*Requires income to be within
20,000 if income above 130,000;
      end;

if fam_size ne peopl then delete;          *Requires household size to be the
same;
if under18 ne WAsurvey18 then delete;      *Requires households to have
children under 18;

_type_=0; *Create variable for merging with output of a proc sum;


*Count number and total weight of candidate donors;

proc summary data =threea;
var finlwt21;
output out=threeb sum=wgttotal;
run;


*Select 7 donors for multiple imputation;

%do i = 1 %to 7 %by 1;

*Calc a score equal to wgt of each donor relative to all donors in this
group. Spread the scores over the interval 0 to 1 with width equal
to each donor's relative weight. Pick a random number between 0 and 1
and select a donor;
*Repeat 7 times;

data three;
merge threea threeb;
by _type_;
rename _freq_ = total;
drop _type_;
score = finlwt21/wgttotal;
if _n_=1 then
a = ranuni(0);
```

```
retain a;
retain score0 score1 0;
score0 = score1;
score1 + score;
newid&i = newid;
Keep ID newid&i _freq_;
if score0< a <= score1 then output three;


*Add imputed value to 1st round data set;

Data  mdimpute.ADImpute&x;
merge mdimpute.ADImpute&x three;
by ID;
I_round='a';
attrib I_round label ='Imputation Round' format =$1.0;
%end; *Ends multiple imputation rounds;
%let a=%eval(&a+1); *Increments 'a' to next missing obs;
%end;
%mend Impute;
%Impute(a=1);
run;

*-------------------------------------------------------------------------;
*SECOND ROUND OF IMPUTATON - Relax family size and income restrictions;
*-------------------------------------------------------------------------;

*Select obs where less than 1 match was found in first round;

Data b;
set mdimpute.ADImpute&x;
Where total lt 1;
run;
Data bmissing;
merge b(in=in1) missing(in=in2) ;
by id; if in1;
run;

*Count the number of obs missing data set;

proc means data = bmissing noprint;
var flag;
output out=bmiss sum=;
run;

*Delete file which will contain imputed values;

Proc delete data=mdimpute.BDImpute&x;
run;

*Create file which will contain imputed values;

Data mdimpute.BDImpute&x;
set bmissing;
keep ID;
Proc sort;
by ID;
```

```
run;

*Create macro variable equal to number of obs with missing AV;

Data bnum;
set bmiss;
call symput('bnum',left(_freq_));

*Macro function to do imputation;

%macro BImpute(b=);

*Loop which starts with first obs and ends with last one;

%do %until (&b gt &bnum);

*Pull the next obs with missing data;

Data two;
Set bmissing;
If _N_ =&b;

*Merge Donor data set with missing data set;
*Keep only Obs from Donor data set with similar characteristics;

Data threea;
Merge dfmly&x Two; By Flag;
if own ne q3p2 and q3p2 in (1,2,3)
then delete;                                *Requires exact match for tenure;

if bld ne bldpopsurvey and bldpopsurvey in (1,3,5)
then delete;                                *Requires exact match for bld type;

if over64popsur ne over64 then delete; *Requires presence of person over64;

if hhinc <100000 then do;
   if fincbtax gt (hhinc +15000)
   or fincbtax lt (hhinc-15000) then delete; *Requires income to be within
15,000 if income below 100,000;
   end;
if 100000<= hhinc <= 130000 then do;
      if fincbtax gt (hhinc+25000)
      or fincbtax lt (hhinc-25000) then delete;*Requires income to be within
25,000 if income above 100,000 and below 130,000;
      end;
if hhinc > 130000 then do;
      if fincbtax gt (hhinc+35000)
      or fincbtax lt (hhinc-35000) then delete;*Requires income to be within
35,000 if income above 130,000;
      end;

if fam_size gt (peopl+1)
or fam_size lt (peopl-1) then delete; *Requires household size to be within
one;

if under18 ne WAsurvey18 then delete;      *Requires households to have
children under 18;
```

```
_type_=0; *Create variable for merging with output of a proc sum;


*Count number and total weight of candidate donors;

proc summary data =threea;
var finlwt21;
output out=threeb sum=wgttotal;
run;


*Select 7 donors for multiple imputation;

%do i = 1 %to 7 %by 1;

*Calc a score equal to wgt of each donor relative to all donors in this
group. Spread the scores over the interval 0 to 1 with width equal
to each donor's relative weight. Pick a random number between 0 and 1
and select a donor;
*Repeat 7 times;

data three;
merge threea threeb;
by _type_;
rename _freq_ = total;
drop _type_;
score = finlwt21/wgttotal;
if _n_=1 then
a = ranuni(0);
retain a;
retain score0 score1 0;
score0 = score1;
score1 + score;
newid&i = newid;
Keep ID newid&i _freq_;
if score0< a <= score1 then output three;


*Add imputed value to 2st round data set;

Data  mdimpute.BDImpute&x;
merge mdimpute.BDImpute&x three;
by ID;
I_round='b';
attrib I_round label ='Imputation Round' format =$1.0;
%end; *Ends multiple imputation rounds;
%let b=%eval(&b+1);*Increments 'b' to next missing obs;
%end;
%mend BImpute;
%BImpute(b=1);
run;

*-------------------------------------------------------------------------;
*THIRD ROUND OF IMPUTATON - eliminate bld type restriction
and relax family size restriction;
*-------------------------------------------------------------------------;
```

```
*Select obs where less than 1 match was found in second round;

Data c;
set mdimpute.BDImpute&x;
Where total lt 1;
run;
Data cmissing;
merge c(in=in1) missing(in=in2) ;
by id; if in1;
run;

*Count the number of obs with missing AV;

proc means data = cmissing noprint;
var flag;
output out=cmiss sum=total;
run;

*Delete file which will contain imputed values;

Proc delete data=mdimpute.CDImpute&x;
run;

*Create file which will contain imputed values;

Data mdimpute.CDImpute&x;
set cmissing;
keep ID;
Proc sort;
by ID;
run;

*Create macro variable equal to number of obs with missing data;

Data bnum;
set cmiss;
call symput('cnum',left(_freq_));

*Macro function to do imputation;

%macro CImpute(c=);

*Loop which starts with first obs with missing AV and ends with last one;

%do %until (&c gt &cnum);

*Pull the next obs with missing data;

Data two;
Set cmissing;
If _N_ =&c;

*Merge Donor data set with obs with missing data;
*Keep only Obs from Donor data set with similar characteristics;

Data threea;
```

```
Merge dfmly&x Two; By Flag;
if own ne q3p2 and q3p2 in (1,2,3)
then delete;                                *Requires exact match for tenure;

*if bld ne bldpopsurvey and bldpopsurvey in (1,3,5)
then delete;                                *Requires exact match for bld type;

if over64popsur ne over64 then delete; *Requires presence of person over64;

if hhinc <100000 then do;
   if fincbtax gt (hhinc +15000)
   or fincbtax lt (hhinc-15000) then delete; *Requires income to be within
15,000 if income below 100,000;
   end;
if 100000<= hhinc <= 130000 then do;
      if fincbtax gt (hhinc+25000)
      or fincbtax lt (hhinc-25000) then delete;*Requires income to be within
25,000 if income above 100,000 and below 130,000;
      end;
if hhinc > 130000 then do;
      if fincbtax gt (hhinc+35000)
      or fincbtax lt (hhinc-35000) then delete;*Requires income to be within
35,000 if income above 130,000;
      end;

if fam_size gt (peopl+2)
or fam_size lt (peopl-2) then delete; *Requires household size to be within
two;

if under18 ne WAsurvey18 then delete;      *Requires households to have
children under 18;

_type_=0; *Create variable for merging with output of a proc sum;


*Count number and total weight of candidate donors;

proc summary data =threea;
var finlwt21;
output out=threeb sum=wgttotal;
run;


*Select 7 donors for multiple imputation;

%do i = 1 %to 7 %by 1;

*Calc a score equal to wgt of each donor relative to all donors in this
group. Spread the scores over the interval 0 to 1 with width equal
to each donor's relative weight. Pick a random number between 0 and 1
and select a donor;
*Repeat 7 times;

data three;
merge threea threeb;
by _type_;
rename _freq_ = total;
```

```
drop _type_;
score = finlwt21/wgttotal;
if _n_=1 then
a = ranuni(0);
retain a;
retain score0 score1 0;
score0 = score1;
score1 + score;
newid&i = newid;
Keep ID newid&i _freq_;
if score0< a <= score1 then output three;

*Add imputed value to 3rd round data set;

Data  mdimpute.CDImpute&x;
merge mdimpute.CDImpute&x three;
by ID;
I_round='c';
attrib I_round label ='Imputation Round' format =$1.0;
%end; *Ends multiple imputation rounds;
%let c=%eval(&c+1);*Increments 'c' to next missing obs;
%end;
%mend CImpute;
%CImpute(c=1);
run;
%end;
%mend Loop;
%Loop;
run;


*-------------------------------------------------------------------------;
*Combine the data sets from the 3 imputation rounds;
*Drop any obs without a match;
*-------------------------------------------------------------------------;

%macro loop1;
%do x=1 %to 4 %by 1;
*Delete file which will contain imputed values;
Proc delete data=mdimpute.DImpute&x;
run;

data mdimpute.DImpute&x;
merge mdimpute.ADImpute&x mdimpute.BDImpute&x mdimpute.CDImpute&x;
by id;
if newid1 = . then delete;
run;
%end;
%mend Loop1;
%loop1;

*-------------------------------------------------------------------------;
*Determine number of quarters that each id has a match;
*-------------------------------------------------------------------------;

data mdimpute.dqtrsmatched;
merge mdimpute.dimpute1(in=a) mdimpute.dimpute2(in=b)
```

```
mdimpute.dimpute3(in=c) mdimpute.dimpute4(in=d);
by id;
if a then qtr1=1; else qtr1=0;
if b then qtr2=1; else qtr2=0;
if c then qtr3=1; else qtr3=0;
if d then qtr4=1; else qtr4=0;
Dqtrs=qtr1+qtr2+qtr3+qtr4;
keep id Dqtrs;
proc sort data=mdimpute.dqtrsmatched;
by id;
run;

options notes source source2 errors=20;

*------------------------------------------------------------------------;
*Check to see if a newid is missing when other iterations contain values;
*------------------------------------------------------------------------;

proc delete data=test;
run;
%macro loop2;
%do w=1 %to 4 %by 1;
data zzz;
set mdimpute.AdImpute&w;
drop I_round&w;
if newid1 = . and sum(newid2, newid3, newid4, newid5, newid6,
newid7)>0 then output;
if newid2 = . and sum(newid1, newid3, newid4, newid5, newid6,
newid7)>0 then output;
if newid3 = . and sum(newid2, newid1, newid4, newid5, newid6,
newid7)>0 then output;
if newid4 = . and sum(newid2, newid3, newid1, newid5, newid6,
newid7)>0 then output;
if newid5 = . and sum(newid2, newid3, newid4, newid1, newid6,
newid7)>0 then output;
if newid6 = . and sum(newid2, newid3, newid4, newid5, newid1,
newid7)>0 then output;
if newid7 = . and sum(newid2, newid3, newid4, newid5, newid6,
newid1)>0 then output;
run;
proc append base = test Data=zzz;
run;
%end;
%mend loop2;
%loop2;
```

```
*************************************************************************;
* program:    Hot decking Interview CEX to WAPOP survey multiple imp     ;
*             with donor wgt adj.sas                                     ;
* programmer: Rick Peterson                                              ;
* project:    Washington Excise Tax Microsimulation Model                ;
* date:       March 21, 2002                                            ;
*                                                                        ;
* purpose:    Match households in WA pop survey with similar households  ;
*             in Consumer Expenditure Survey. Matching variables: Income,;
*             housing tenure, housing building type, presence of person  ;
*             over 64 in household, presence of person under 18 in       ;
*             household, and family size.                                ;
*                                                                        ;
*------------------------------------------------------------------------;
* libraries: popsur - location of WAPOP survey data                      ;
*            miimpute - destination of matching files                    ;
*                                                                        ;
*------------------------------------------------------------------------;
* incoming:   From Interview CEX: fmly99.1, fmly99.2, fmly99.3 fmly99.4  ;
*             From WA population survey: sps00f04                        ;
*                                                                        ;
*------------------------------------------------------------------------;
* formats:    Format for WAPOP survey                                      ;
*                                                                        ;
*------------------------------------------------------------------------;
* outgoing:   IImpute1, IImpute2, IImpute3, IImpute4: containing household;
*             matches.                                                   ;
*             iqtrsmatched: containing number of quarters matched for    ;
*             each household - used in later program to adjust consumption;
*             data.                                                      ;
*                                                                        ;
*------------------------------------------------------------------------;
* reports:    None                                                       ;
*                                                                        ;
*------------------------------------------------------------------------;
* changes:                                                               ;
*                                                                        ;
*------------------------------------------------------------------------;
* notes:      Each Washington household is matched to a CEX household for ;
*             each of the 4 quarter of CEX data.  Matching is done in    ;
*             three rounds.  If a match is not found then matching       ;
*             criteria are relaxed for next round.  When more than one   ;
*             donor is available then the donor is selected randomly.    ;
*             The likelihood of donor selection is proportional to the   ;
*             relative weight of the donor within the group of potential ;
*             donors.                                                    ;
*                                                                        ;
*                                                                        ;
*************************************************************************;

*------------------------------------------------------------------------;
*Read in donor pool from Interview data sets;
*------------------------------------------------------------------------;

%let y = 99;
filename ifmly1 "c:\data\interview survey\fmlyi&y.1x.txt";
filename ifmly2 "c:\data\interview survey\fmlyi&y.2.txt";
```

```
filename ifmly3 "c:\data\interview survey\fmlyi&y.3.txt";
filename ifmly4 "c:\data\interview survey\fmlyi&y.4.txt";

%macro loop1;
%do x= 1 %to 4 %by 1;
data ifmly&x;
   infile ifmly&x lrecl=3461;
   input @1 newid 8.
            @331 finlwt21 11.3
            @216 cutenure $1.
            @242 Fam_size 2.
         @633 perslt18 2.
            @636 persot64 2.
            @245 Fam_type $1.
            @556 no_earner 2.
            @3324 inclass $2.
            @681 respstat $1.
            @11 age_ref 2.
            @303 fincbtax 9.
            @61 building $2.
            @351 fnonfrmx 9.
            @274 ffrmincx 9.
            @456 inclossa 8.
            @465 inclossb 8.
            @506 jfdstmpa 8.
            ;
*Eliminate obs without complete reporting of income;
if respstat ne '1' then delete;
*Create a variable to merge with WAPOP survey observation;
Flag = 1;
*Create variable to match housing tenure;
if cutenure in ('1', '2','3') then own = 1;
else if cutenure eq '4' then own =2;
else if cutenure eq '5' then own =3;
else own =4;
*Create variable to match building type;
if building in ('01') then bld = 1;
else if building in ('02','03','04','05','06','07','08') then bld = 2;
else if building in ('09') then bld = 3;
else bld = 5;
*Cap household size at 6 people;
if fam_size >5 then fam_size = 6;
*Create variable to match persons in household under 18;
if perslt18>0 then under18 = 1;
   else under18=0;
*Create variable to match persons in household over 64;
   if persot64>0 then over64 = 1;
   else over64 = 0;
*Remove business losses and value of food stamps from income;
   if fnonfrmx ne . and fnonfrmx <0 then fincbtax = fincbtax - fnonfrmx;
   if ffrmincx ne . and ffrmincx <0 then fincbtax = fincbtax - ffrmincx;
   if inclossa ne . and inclossa <0 then fincbtax = fincbtax - inclossa;
   if inclossb ne . and inclossb <0 then fincbtax = fincbtax - inclossb;
   if jfdstmpa ne . then fincbtax = fincbtax - jfdstmpa;
run;
%end;
%mend Loop1;
```

```
%Loop1;

*------------------------------------------------------------------------;
*Read in data from WAPOP survey;
*------------------------------------------------------------------------;

Data one;
set popsur.sps00f04;
*Create variable for presence of person over64 in household;
if age >64 then over64popsur =1;
else over64popsur =0;
keep id age over64popsur;
run;

proc summary data=one;
by id;
var over64popsur;
output out=two sum=;
run;

Data A;
set popsur.sps00f04;
where pnum=1;
Flag = 1;
*Cap household size at 6 people;
if peopl >5 then peopl = 6;
*Create variable to match persons in household under 18;
if chldrn18>0 then WAsurvey18 = 1;
   else WAsurvey18=0;

keep Flag ID pnum age q3p1 q3p2 fnlwgt hhinc peopl WAsurvey18;
run;

*Create "Missing" - the  WAPOP data set with household information but
without consumption data;
*Merge in variable for presence of person over 64;
Data missing;
merge A two;
by id;
if over64popsur >0 then over64popsur=1;
else over64popsur=0;
drop _type_ _freq_;
run;

*------------------------------------------------------------------------;
*FIRST ROUND OF IMPUTATON - Restrict potential donors by tenure, property
type, family size, age and income
*------------------------------------------------------------------------;

*Turn off writing to the SAS log;
options nonotes nosource nosource2 errors=0;

*Create macro loop to find a match for the WAPOP household in each
 of the 4 CEX interview data sets;

%macro loop;
%do x = 1 %to 4;
```

```
*Count the number of obs in missing;

proc means data = missing noprint;
var flag;
output out=amiss sum=;
run;

*Delete file which will contain imputed values;

Proc delete data=miimpute.AIimpute&x;
run;

*Create file which will contain imputed values;

Data miimpute.AIImpute&x;
set missing;
keep ID;
Proc sort;
by ID;
run;

*Create macro variable equal to number of obs with missing CEX data;

Data num;
set amiss;
call symput('num',left(_freq_));

*Macro function to do imputation;

%macro Impute(a=);

*Loop which starts with first obs with missing data and ends with last one;

%do %until (&a gt &num);

*Pull the next obs with missing data;

Data two;
Set missing;
If _N_ =&a;


*Merge Donor data set with missing data set;
*Keep only Obs from Donor data set with similar characteristics;

Data threea;
Merge ifmly&x Two; By Flag;
if own ne q3p2 and q3p2 in (1,2,3)
then delete;                                *Requires exact match for tenure;
if bld ne q3p1 and q3p1 in (1,2,3,5)
then delete;                                *Requires exact match for bld type;

if over64popsur ne over64 then delete; *Requires presence of person over64;

if hhinc <100000 then do;
   if fincbtax gt (hhinc +7500)
```

```
   or fincbtax lt (hhinc-7500) then delete; *Requires income to be within
7,500 if income below 100,000;
   end;
if 100000<= hhinc <= 130000 then do;
      if fincbtax gt (hhinc+12500)
      or fincbtax lt (hhinc-12500) then delete;*Requires income to be within
12,500 if income above 100,000 and below 130,000;
      end;
if hhinc > 130000 then do;
      if fincbtax gt (hhinc+20000)
      or fincbtax lt (hhinc-20000) then delete;*Requires income to be within
20,000 if income above 130,000;
      end;

if fam_size ne peopl then delete;          *Requires household size to be the
same;
if under18 ne WAsurvey18 then delete;      *Requires households to have
children under 18;

_type_=0; *Create variable for merging with output of a proc sum;


*Count number and total weight of candidate donors;

proc summary data =threea;
var finlwt21;
output out=threeb sum=wgttotal;
run;


*Select 7 donors for multiple imputation;

%do i = 1 %to 7 %by 1;

*Calc a score equal to wgt of each donor relative to all donors in this
group. Spread the scores over the interval 0 to 1 with width equal
to each donor's relative weight. Pick a random number between 0 and 1
and select a donor;
*Repeat 7 times;

data three;
merge threea threeb;
by _type_;
rename _freq_ = total;
drop _type_;
score = finlwt21/wgttotal;
if _n_=1 then
a = ranuni(0);
retain a;
retain score0 score1 0;
score0 = score1;
score1 + score;
newid&i = newid;
Keep ID newid&i _freq_;
if score0< a <= score1 then output three;

*Add imputed value to 1st round data set;
```

```
Data  miimpute.AIImpute&x;
merge miimpute.AIImpute&x three;
by ID;
I_round='a';
attrib I_round label ='Imputation Round' format =$1.0;
%end; *Ends multiple imputation rounds;
%let a=%eval(&a+1); *Increments 'a' to next missing obs;
%end;
%mend Impute;
%Impute(a=1);
run;

*-------------------------------------------------------------------------;
*SECOND ROUND OF IMPUTATON - Relax family size and income restrictions
*-------------------------------------------------------------------------;
*Select obs where less than 1 match was found in first round;

Data b;
set miimpute.AIImpute&x;
Where total lt 1;
run;
Data bmissing;
merge b(in=in1) missing(in=in2) ;
by id; if in1;
run;

*Count the number of obs missing data set;

proc means data = bmissing noprint;
var flag;
output out=bmiss sum=;
run;

*Delete file which will contain imputed values;

Proc delete data=miimpute.BIImpute&x;
run;

*Create file which will contain imputed values;

Data miimpute.BIImpute&x;
set bmissing;
keep ID;
Proc sort;
by ID;
run;

*Create macro variable equal to number of obs with missing AV;

Data bnum;
set bmiss;
call symput('bnum',left(_freq_));

*Macro function to do imputation;

%macro BImpute(b=);
```

```
*Loop which starts with first obs and ends with last one;

%do %until (&b gt &bnum);

*Pull the next obs with missing data;

Data two;
Set bmissing;
If _N_ =&b;

*Merge Donor data set with missing data set;
*Keep only Obs from Donor data set with similar characteristics;

Data threea;
Merge ifmly&x Two; By Flag;
if own ne q3p2 and q3p2 in (1,2,3)
then delete;                                *Requires exact match for tenure;

if bld ne q3p1 and q3p1 in (1,2,3,5)
then delete;                                 *Requires exact match for bld
type;

if over64popsur ne over64 then delete; *Requires presence of person over64;

if hhinc <100000 then do;
   if fincbtax gt (hhinc +15000)
   or fincbtax lt (hhinc-15000) then delete; *Requires income to be within
15,000 if income below 100,000;
   end;
if 100000<= hhinc <= 130000 then do;
      if fincbtax gt (hhinc+25000)
      or fincbtax lt (hhinc-25000) then delete;*Requires income to be within
25,000 if income above 100,000 and below 130,000;
      end;
if hhinc > 130000 then do;
      if fincbtax gt (hhinc+35000)
      or fincbtax lt (hhinc-35000) then delete;*Requires income to be within
35,000 if income above 130,000;
      end;

if fam_size gt (peopl+1)
or fam_size lt (peopl-1) then delete; *Requires household size to be within
one;

if under18 ne WAsurvey18 then delete;      *Requires households to have
children under 18;

_type_=0; *Create variable for merging with output of a proc sum;


*Count number and total weight of candidate donors;

proc summary data =threea;
var finlwt21;
output out=threeb sum=wgttotal;
run;
```

```
*Select 7 donors for multiple imputation;

%do i = 1 %to 7 %by 1;

*Calc a score equal to wgt of each donor relative to all donors in this
group. Spread the scores over the interval 0 to 1 with width equal
to each donor's relative weight. Pick a random number between 0 and 1
and select a donor;
*Repeat 7 times;

data three;
merge threea threeb;
by _type_;
rename _freq_ = total;
drop _type_;
score = finlwt21/wgttotal;
if _n_=1 then
a = ranuni(0);
retain a;
retain score0 score1 0;
score0 = score1;
score1 + score;
newid&i = newid;
Keep ID newid&i _freq_;
if score0< a <= score1 then output three;


*Add imputed value to 2st round data set;

Data  miimpute.BIImpute&x;
merge miimpute.BIImpute&x three;
by ID;
I_round='b';
attrib I_round label ='Imputation Round' format =$1.0;
%end; *Ends multiple imputation rounds;
%let b=%eval(&b+1);*Increments 'b' to next missing obs;
%end;
%mend BImpute;
%BImpute(b=1);
run;

*-------------------------------------------------------------------------;
*THIRD ROUND OF IMPUTATON - eliminate bld type restriction
and relax family size restriction;
*-------------------------------------------------------------------------;

*Select obs where less than 1 match was found in second round;

Data c;
set miimpute.BIImpute&x;
Where total lt 1;
run;
Data cmissing;
merge c(in=in1) missing(in=in2) ;
by id; if in1;
```

```
run;

*Count the number of obs with missing AV;

proc means data = cmissing noprint;
var flag;
output out=cmiss sum=total;
run;

*Delete file which will contain imputed values;

Proc delete data=miimpute.CIImpute&x;
run;

*Create file which will contain imputed values;

Data miimpute.CIImpute&x;
set cmissing;
keep ID;
Proc sort;
by ID;
run;

*Create macro variable equal to number of obs with missing data;

Data bnum;
set cmiss;
call symput('cnum',left(_freq_));

*Macro function to do imputation;

%macro CImpute(c=);

*Loop which starts with first obs with missing AV and ends with last one;

%do %until (&c gt &cnum);

*Pull the next obs with missing data;

Data two;
Set cmissing;
If _N_ =&c;

*Merge Donor data set with obs with missing data;
*Keep only Obs from Donor data set with similar characteristics;

Data threea;
Merge ifmly&x Two; By Flag;
if own ne q3p2 and q3p2 in (1,2,3)
then delete;                                *Requires exact match for tenure;

*if bld ne q3p1 and q3p1 in (1,2,3,5)
then delete;                                 *Requires exact match for bld
type;

if over64popsur ne over64 then delete; *Requires presence of person over64;
```

```
if hhinc <100000 then do;
   if fincbtax gt (hhinc +15000)
   or fincbtax lt (hhinc-15000) then delete; *Requires income to be within
15,000 if income below 100,000;
   end;
if 100000<= hhinc <= 130000 then do;
      if fincbtax gt (hhinc+25000)
      or fincbtax lt (hhinc-25000) then delete;*Requires income to be within
25,000 if income above 100,000 and below 130,000;
      end;
if hhinc > 130000 then do;
      if fincbtax gt (hhinc+35000)
      or fincbtax lt (hhinc-35000) then delete;*Requires income to be within
35,000 if income above 130,000;
      end;

if fam_size gt (peopl+2)
or fam_size lt (peopl-2) then delete; *Requires household size to be within
two;

if under18 ne WAsurvey18 then delete;      *Requires households to have
children under 18;

_type_=0; *Create variable for merging with output of a proc sum;


*Count number and total weight of candidate donors;

proc summary data =threea;
var finlwt21;
output out=threeb sum=wgttotal;
run;


*Select 7 donors for multiple imputation;

%do i = 1 %to 7 %by 1;

*Calc a score equal to wgt of each donor relative to all donors in this
group. Spread the scores over the interval 0 to 1 with width equal
to each donor's relative weight. Pick a random number between 0 and 1
and select a donor;
*Repeat 7 times;

data three;
merge threea threeb;
by _type_;
rename _freq_ = total;
drop _type_;
score = finlwt21/wgttotal;
if _n_=1 then
a = ranuni(0);
retain a;
retain score0 score1 0;
score0 = score1;
score1 + score;
newid&i = newid;
```

```
Keep ID newid&i _freq_;
if score0< a <= score1 then output three;

*Add imputed value to 3rd round data set;

Data  miimpute.CIImpute&x;
merge miimpute.CIImpute&x three;
by ID;
I_round='c';
attrib I_round label ='Imputation Round' format =$1.0;
%end; *Ends multiple imputation rounds;
%let c=%eval(&c+1);*Increments 'c' to next missing obs;
%end;
%mend CImpute;
%CImpute(c=1);
run;

%end;
%mend Loop;
%Loop;
run;

*-------------------------------------------------------------------------;
*Combine the data sets from the 3 imputation rounds;
*Drop any obs without a match;
*-------------------------------------------------------------------------;

*Delete file which will contain imputed values;
%macro loop1;
%do x=1 %to 4 %by 1;
Proc delete data=miimpute.IImpute&x;
run;

data miimpute.IImpute&x;
merge miimpute.AIImpute&x miimpute.BIImpute&x miimpute.CIImpute&x;
by id;
if newid&x = . then delete;
run;
%end;
%mend Loop1;
%loop1;

*-------------------------------------------------------------------------;
*Determine number of quarters that each id has a match;
*-------------------------------------------------------------------------;

data miimpute.iqtrsmatched;
merge miimpute.iimpute1(in=a) miimpute.iimpute2(in=b)
miimpute.iimpute3(in=c) miimpute.iimpute4(in=d);
by id;
if a then qtr1=1; else qtr1=0;
if b then qtr2=1; else qtr2=0;
if c then qtr3=1; else qtr3=0;
if d then qtr4=1; else qtr4=0;
Dqtrs=qtr1+qtr2+qtr3+qtr4;
keep id Dqtrs;
proc sort data=miimpute.iqtrsmatched;
```

```
by id;
run;

options notes source source2 errors=20;

*-------------------------------------------------------------------------;
*Check to see if a newid is missing when other iterations contain values;
*-------------------------------------------------------------------------;

proc delete data=test;
run;
%macro loop2;
%do w=1 %to 4 %by 1;
data zzz;
set miimpute.AIImpute&w;
if newid1 = . and sum(newid2, newid3, newid4, newid5, newid6,
newid7)>0 then output;
if newid2 = . and sum(newid1, newid3, newid4, newid5, newid6,
newid7)>0 then output;
if newid3 = . and sum(newid2, newid1, newid4, newid5, newid6,
newid7)>0 then output;
if newid4 = . and sum(newid2, newid3, newid1, newid5, newid6,
newid7)>0 then output;
if newid5 = . and sum(newid2, newid3, newid4, newid1, newid6,
newid7)>0 then output;
if newid6 = . and sum(newid2, newid3, newid4, newid5, newid1,
newid7)>0 then output;
if newid7 = . and sum(newid2, newid3, newid4, newid5, newid6,
newid1)>0 then output;
run;
proc append base = test Data=zzz;
run;
%end;
%mend loop2;
%loop2;
```

[i] These programs were adapted from James W. McNally's paper – "Generating Hot-Deck Imputation Estimates: Using SAS for Simple and Multiple Imputation Allocation Routines", PSTC Working Paper #97-12 , Sept 1997, Population Studies and Training Center, Brown University.

Successful Matches of SPS with CEX By Imputation Round

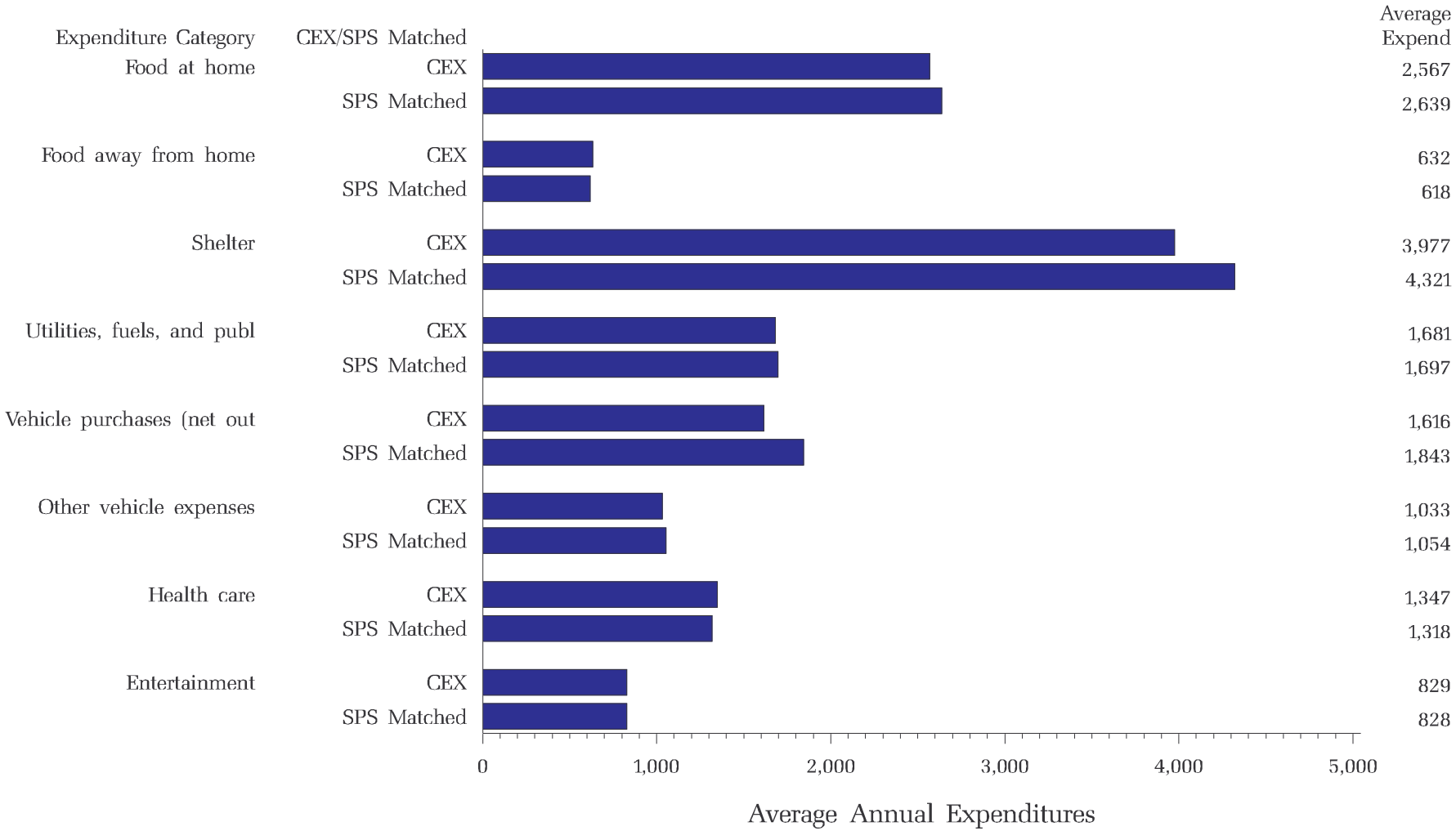
		Imputation Round		
		First Round	Second Round	Third Round
Survey				
Diary 1st Quarter	Number	6,071	467	110
	Percent	91.3	7.0	1.7
Diary 2nd Quarter	Number	5,963	537	155
	Percent	89.6	8.1	2.3
Diary 3rd Quarter	Number	6,071	467	110
	Percent	91.3	7.0	1.7
Diary 4th Quarter	Number	5,963	537	155
	Percent	89.6	8.1	2.3
Interview 1st Quarter	Number	6,315	323	57
	Percent	94.3	4.8	0.9
Interview 2nd Quarter	Number	6,397	227	58
	Percent	95.7	3.4	0.9
Interview 3rd Quarter	Number	6,315	323	57
	Percent	94.3	4.8	0.9
Interview 4th Quarter	Number	6,397	227	58
	Percent	95.7	3.4	0.9

Appendix 4

Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

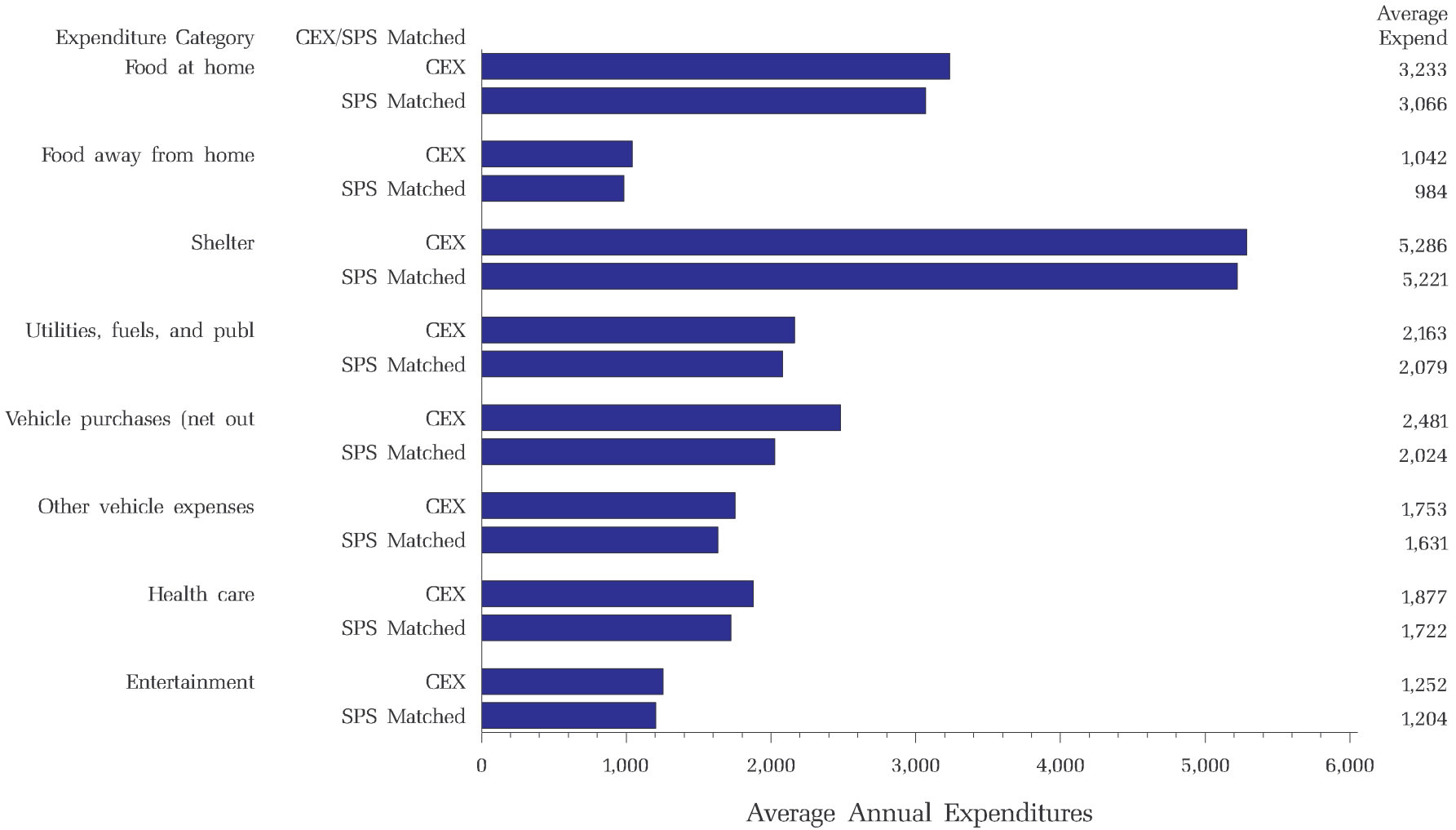
1999 HOUSEHOLD TOTAL INCOME = $0 to $20,000



Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

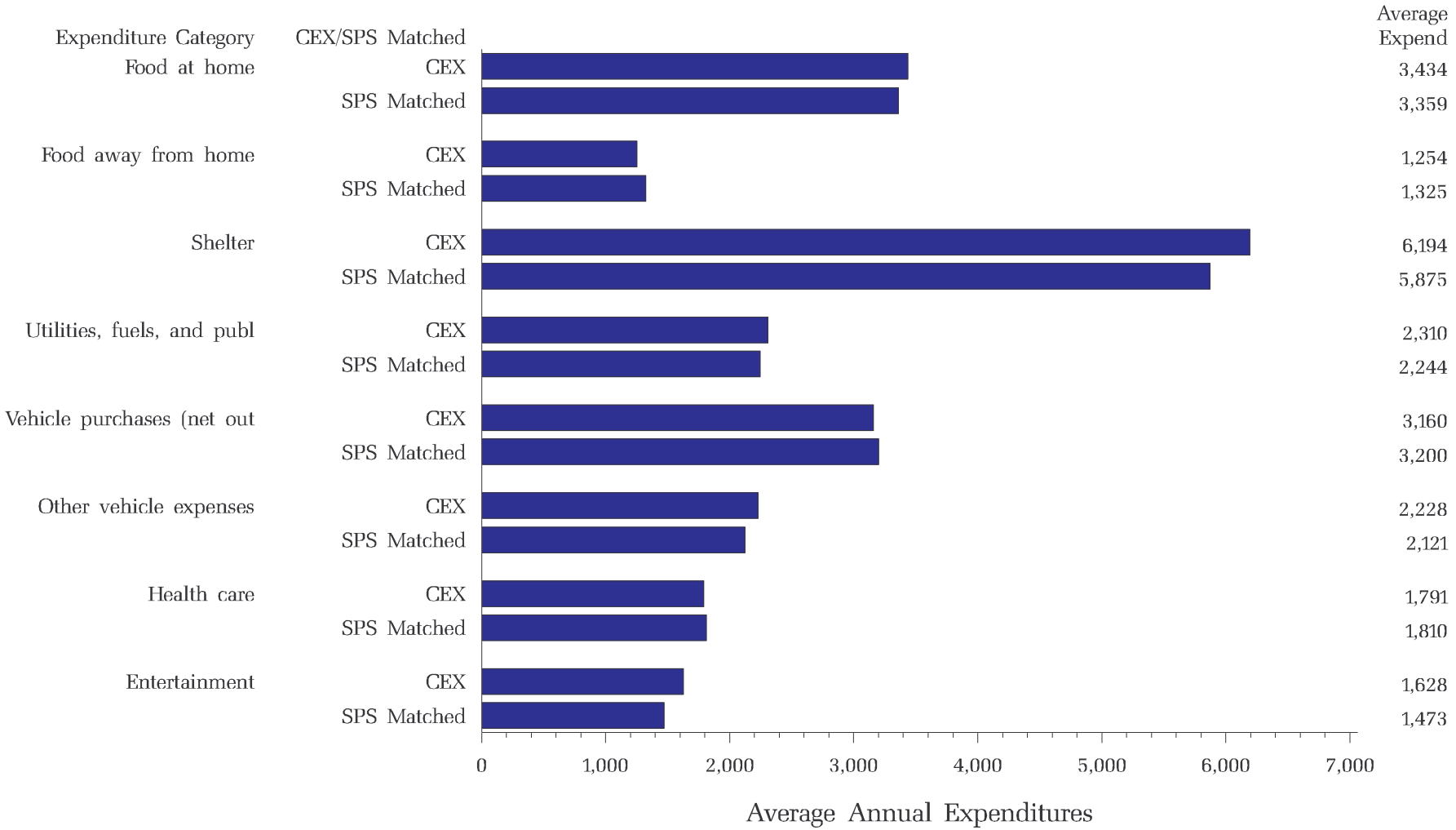
1999 HOUSEHOLD TOTAL INCOME = $20,000 to $30,000



Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

1999 HOUSEHOLD TOTAL INCOME = $30,000 to $40,000

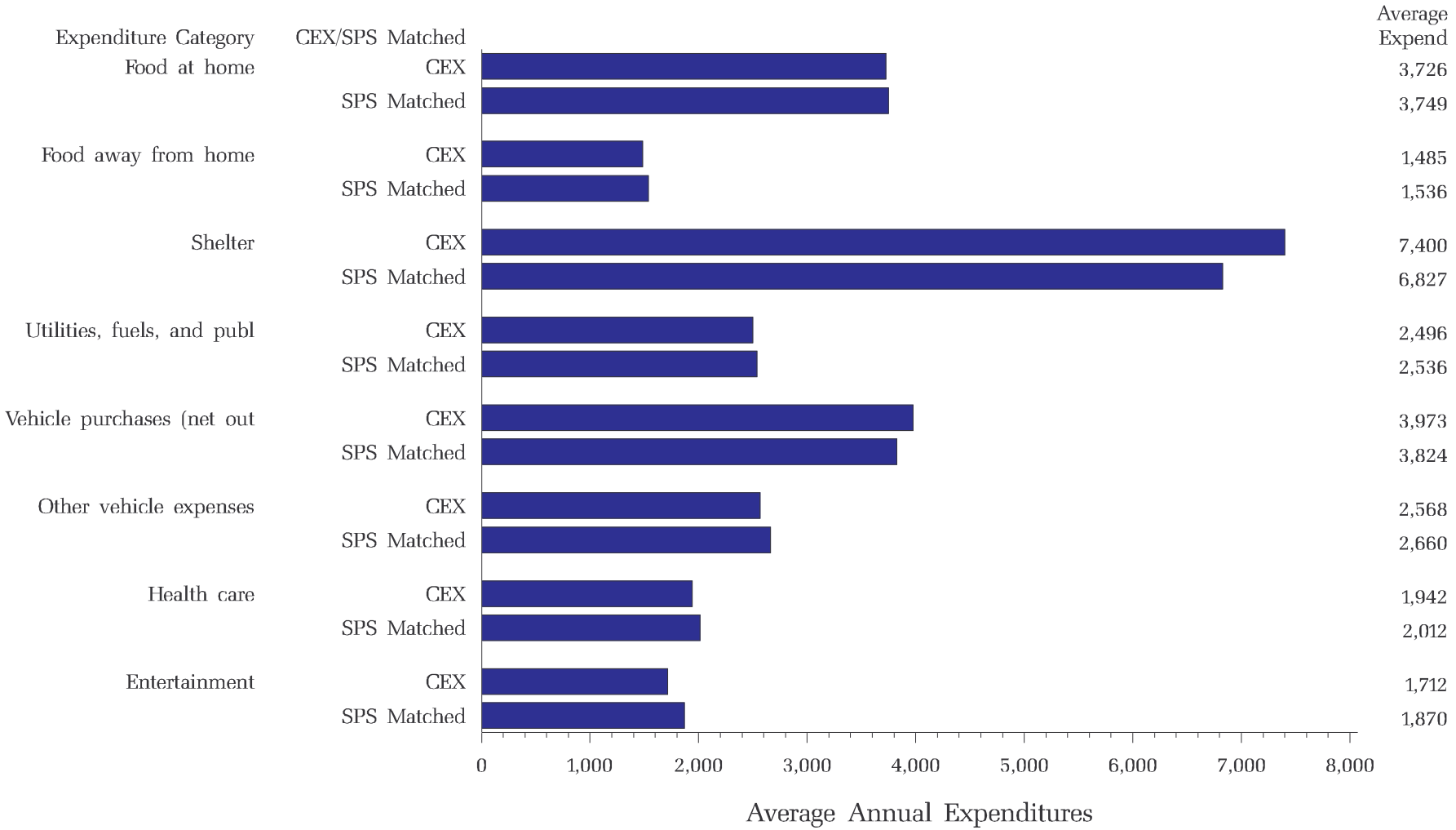


Appendix 4

Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

1999 HOUSEHOLD TOTAL INCOME = $40,000 to $50,000

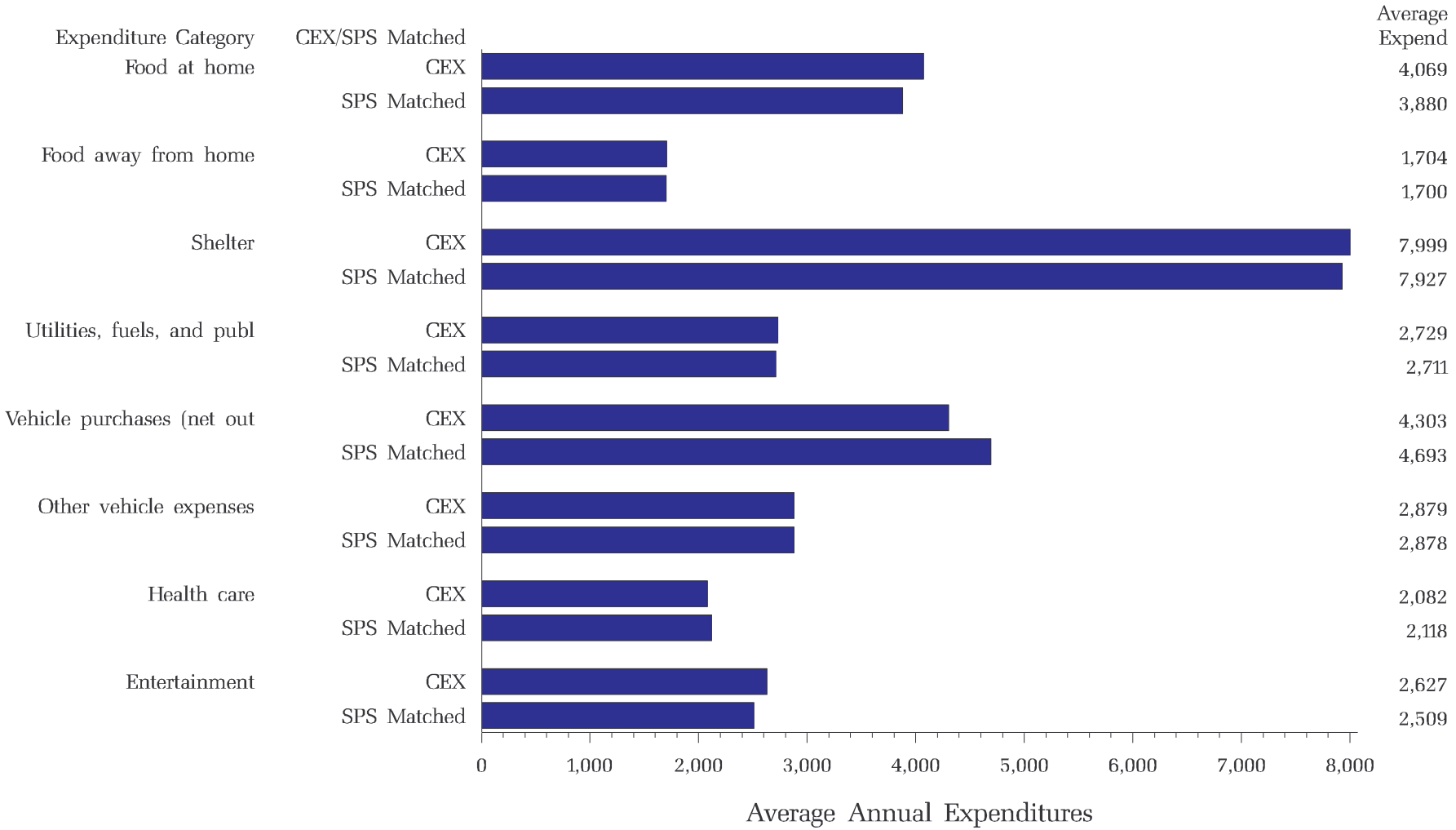


Appendix 4

Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

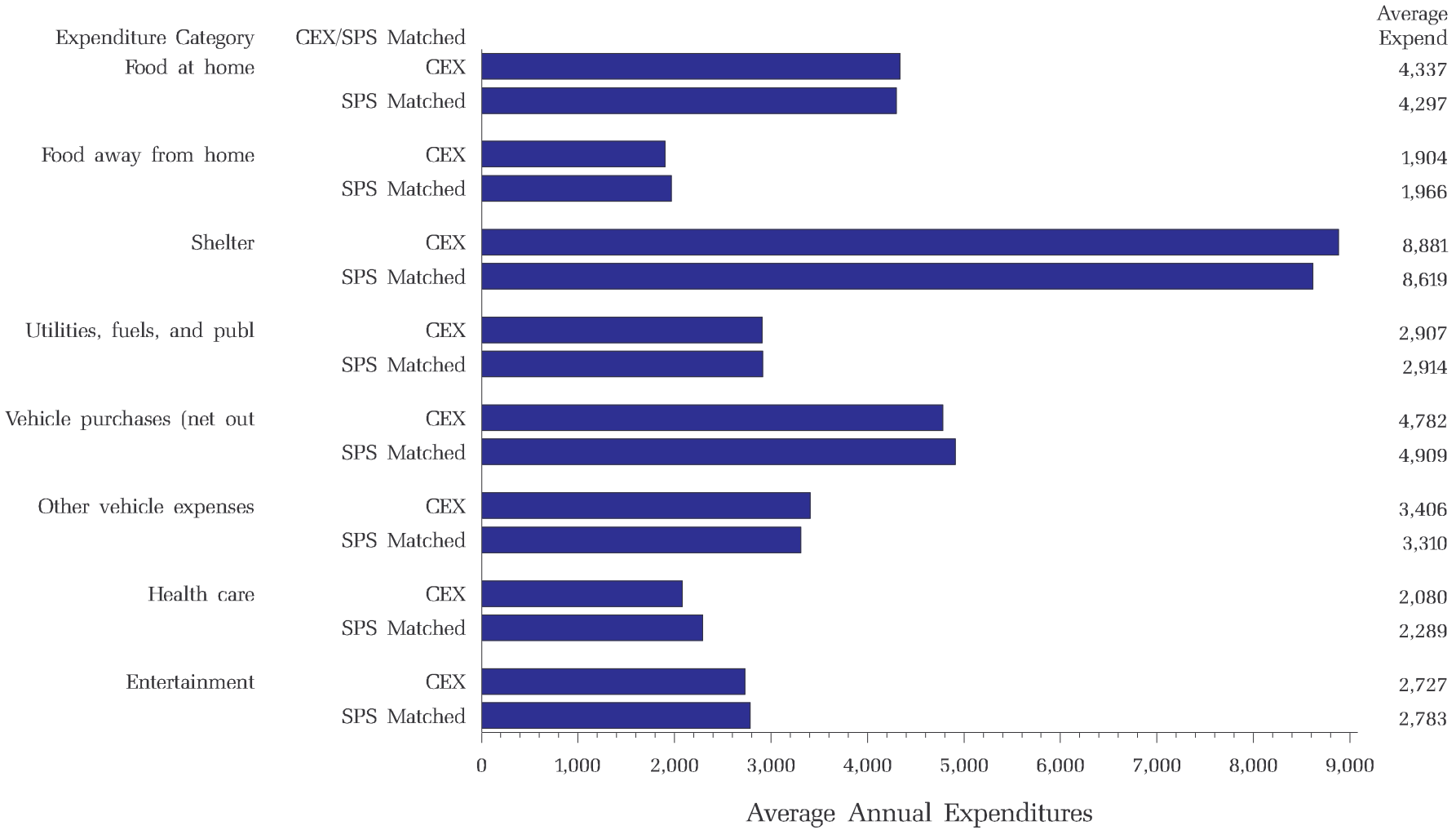
1999 HOUSEHOLD TOTAL INCOME = $50,000 to $60,000



Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

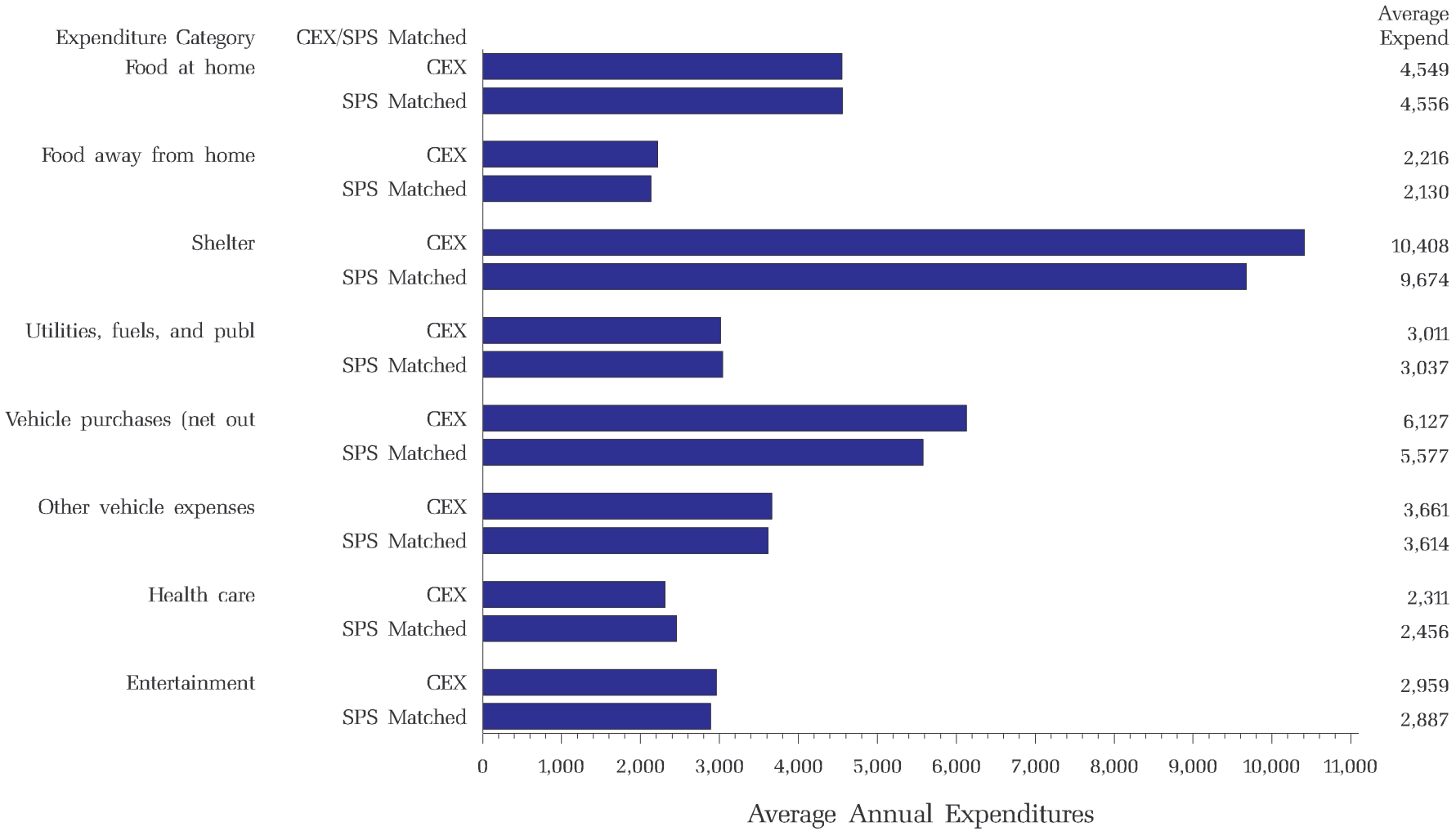
1999 HOUSEHOLD TOTAL INCOME = $60,000 to $70,000



Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

1999 HOUSEHOLD TOTAL INCOME = $70,000 to $80,000

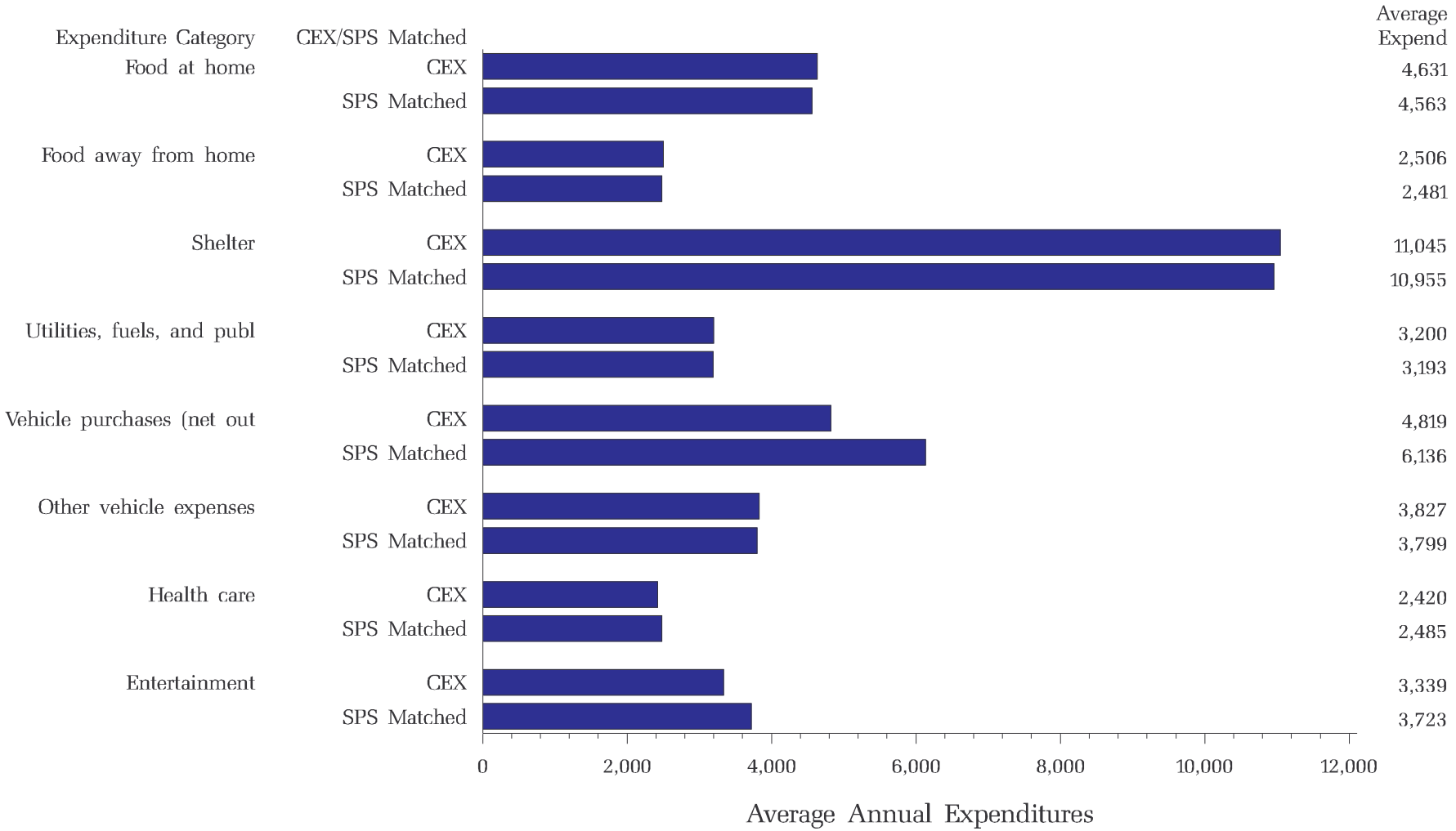


Appendix 4

Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

1999 HOUSEHOLD TOTAL INCOME=$80,000 to $100,000

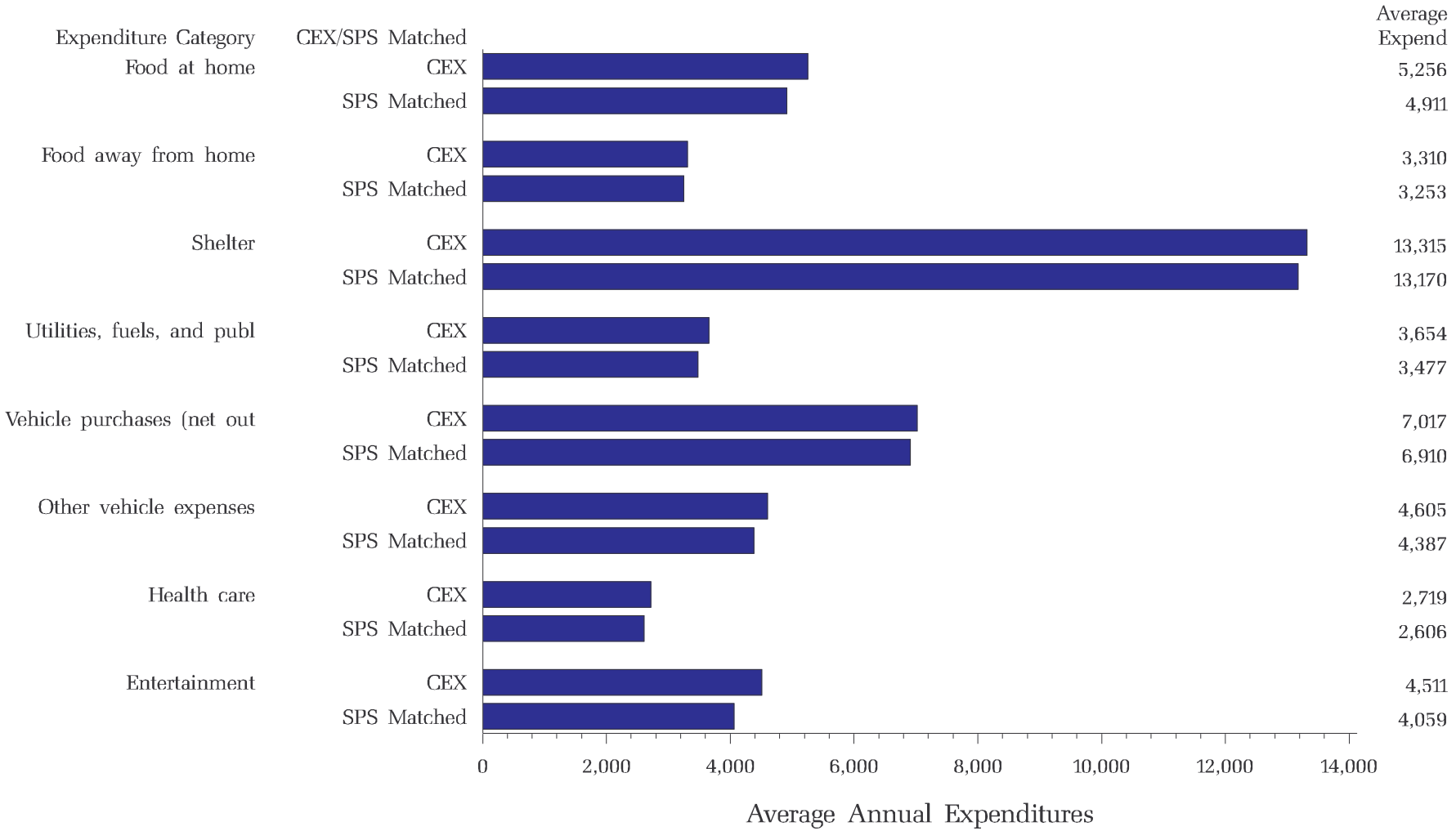


Appendix 4

Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

1999 HOUSEHOLD TOTAL INCOME=$100,000 to $130,000

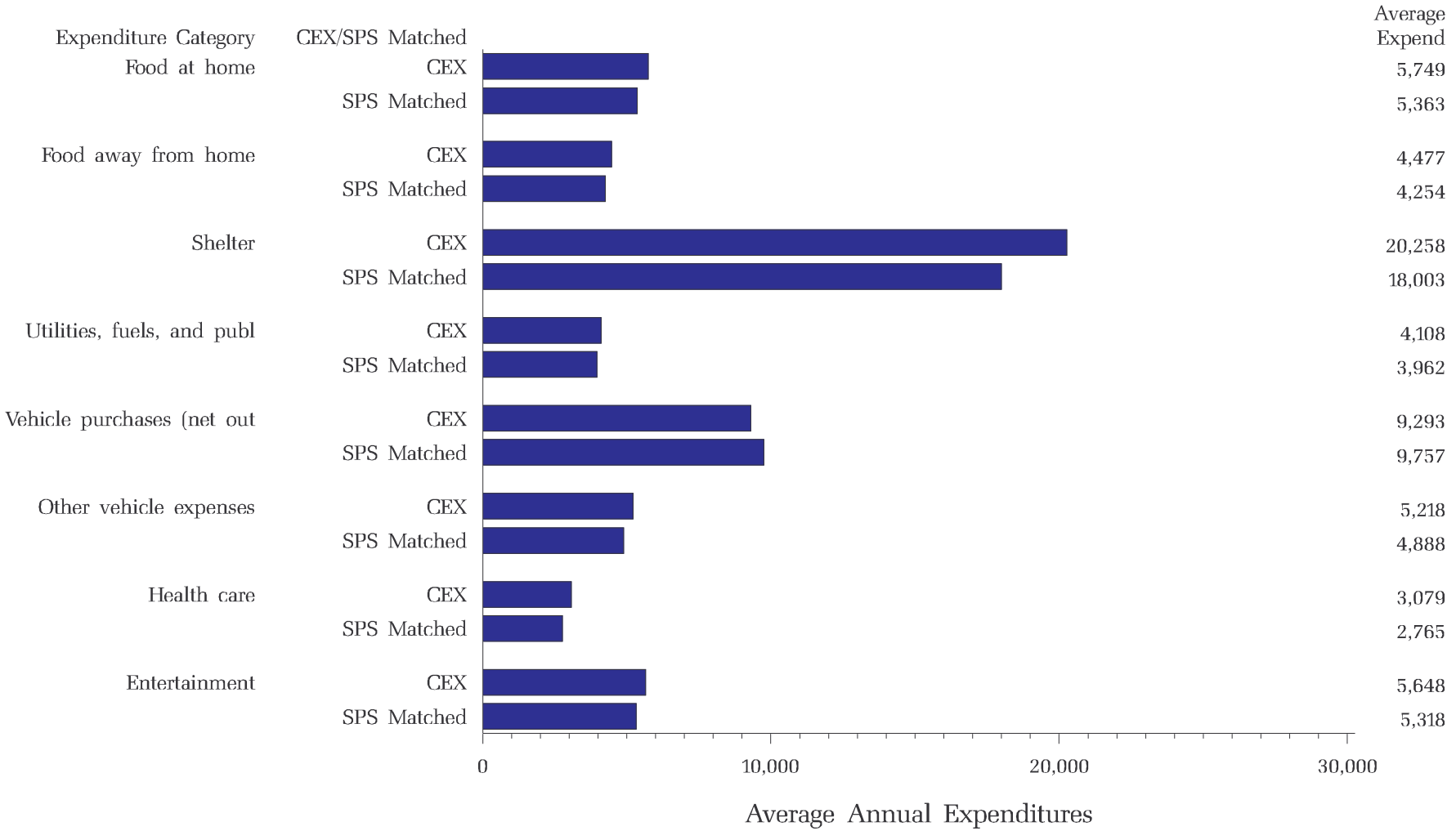


Appendix 4

Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

1999 HOUSEHOLD TOTAL INCOME = Over $130,000

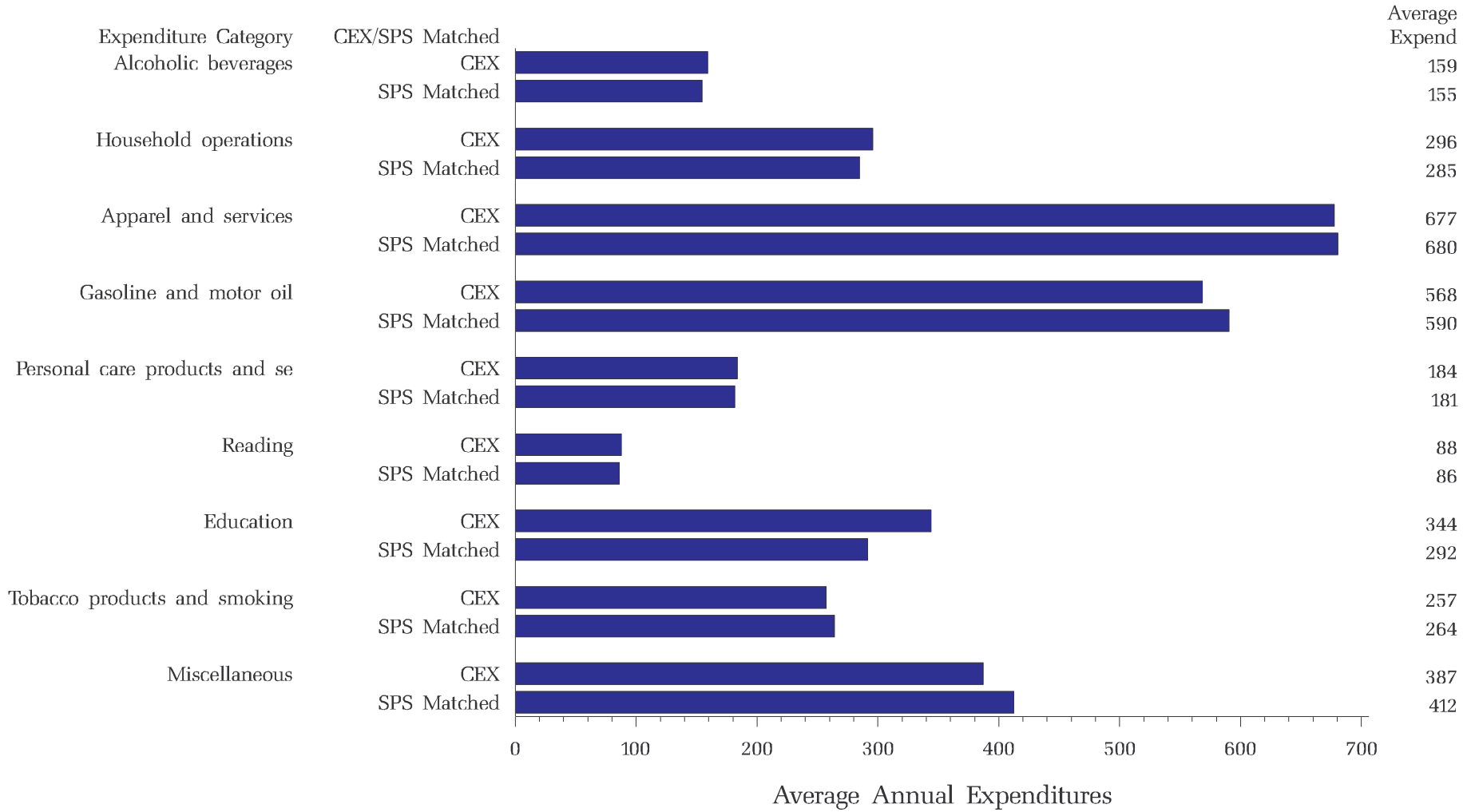


Appendix 4

Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

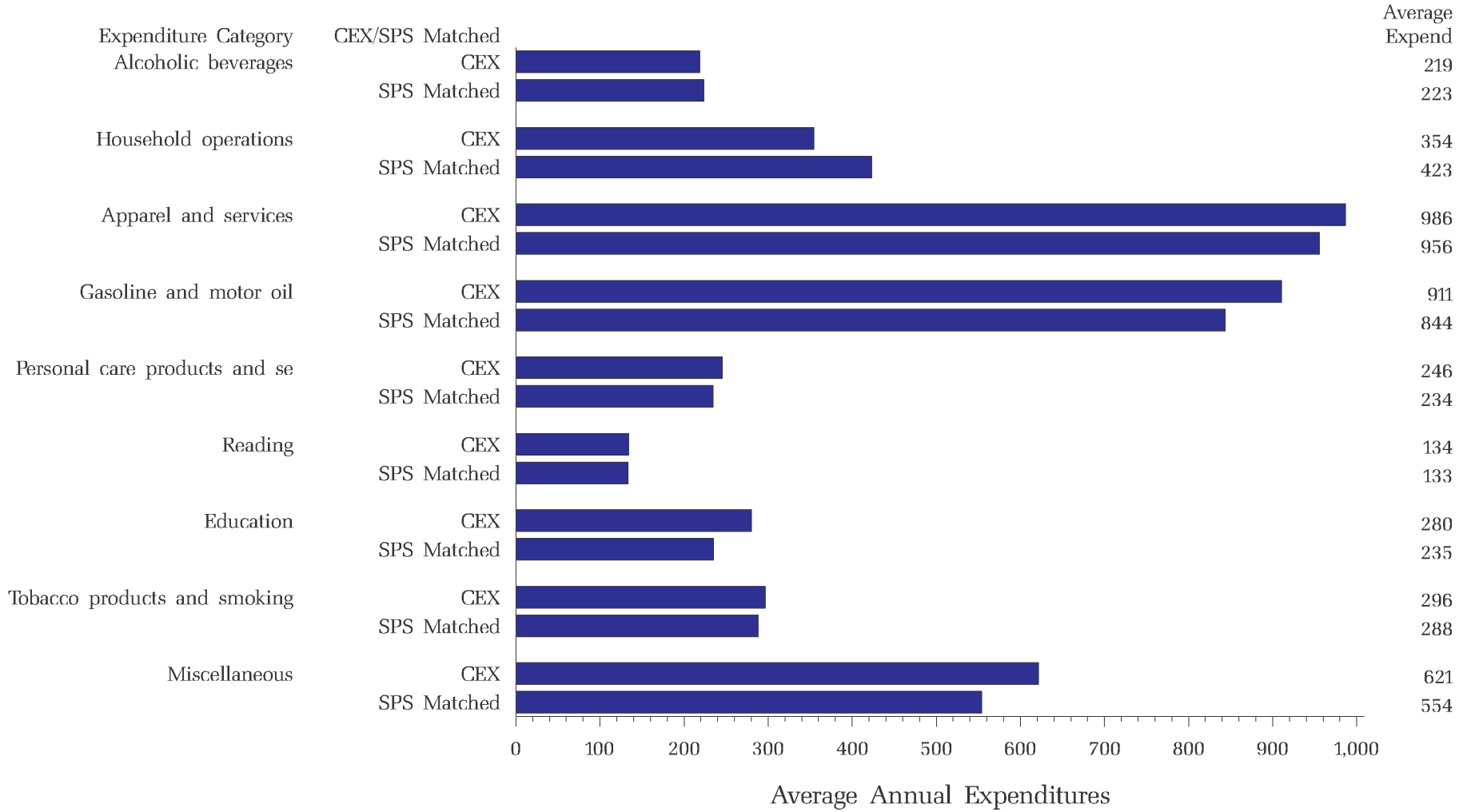
1999 HOUSEHOLD TOTAL INCOME = $0 to $20,000



Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

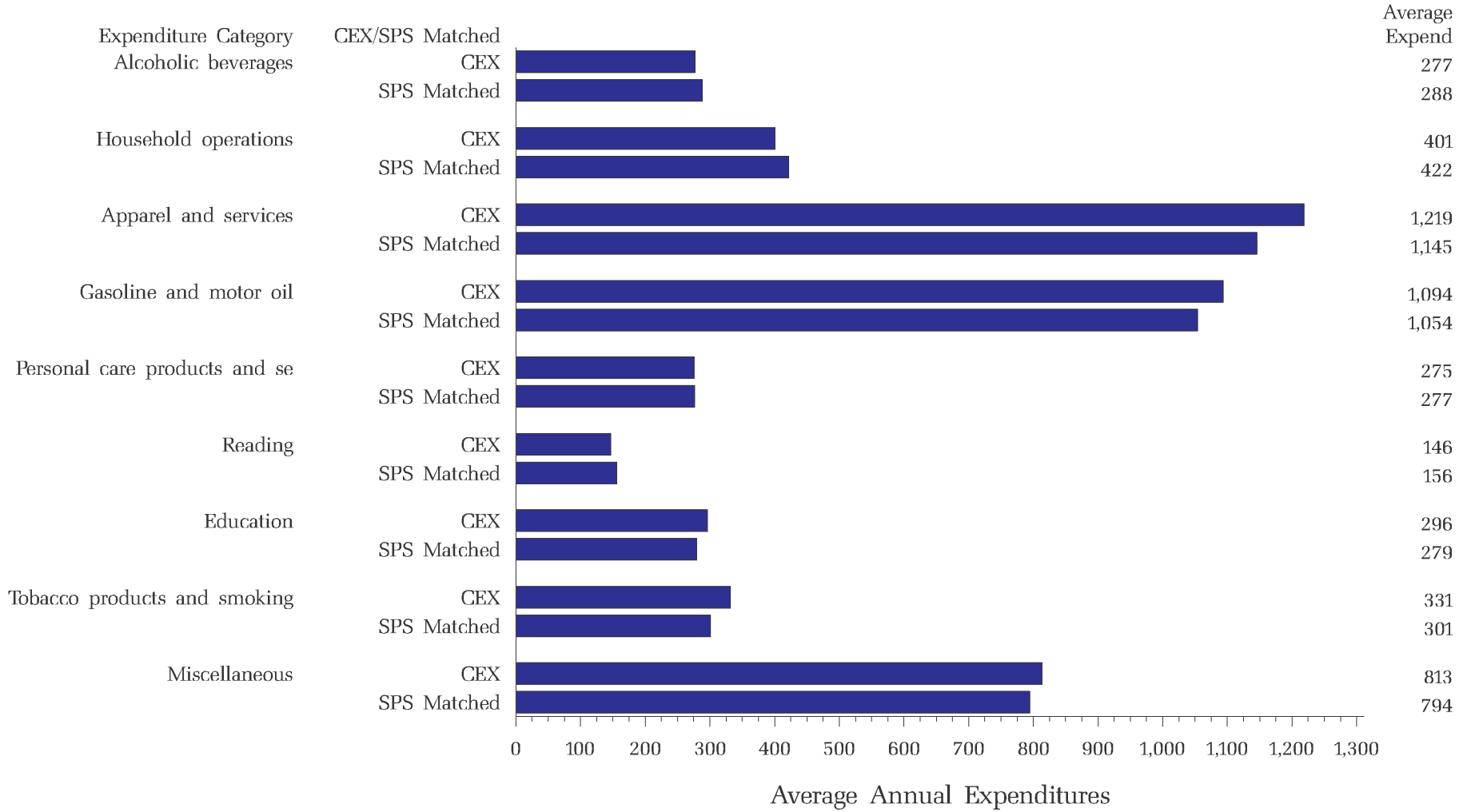
1999 HOUSEHOLD TOTAL INCOME = $20,000 to $30,000



Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

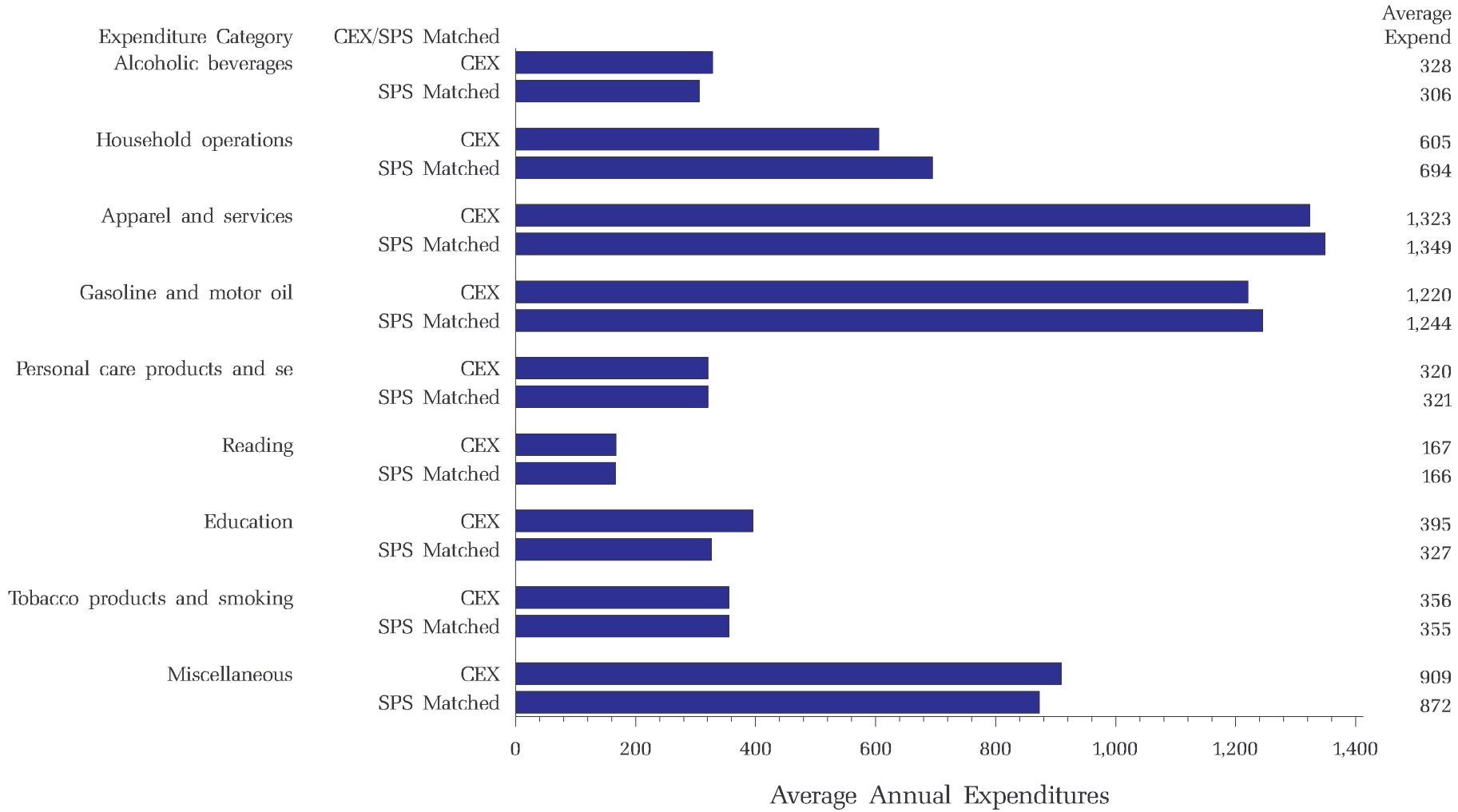
1999 HOUSEHOLD TOTAL INCOME = $30,000 to $40,000



Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

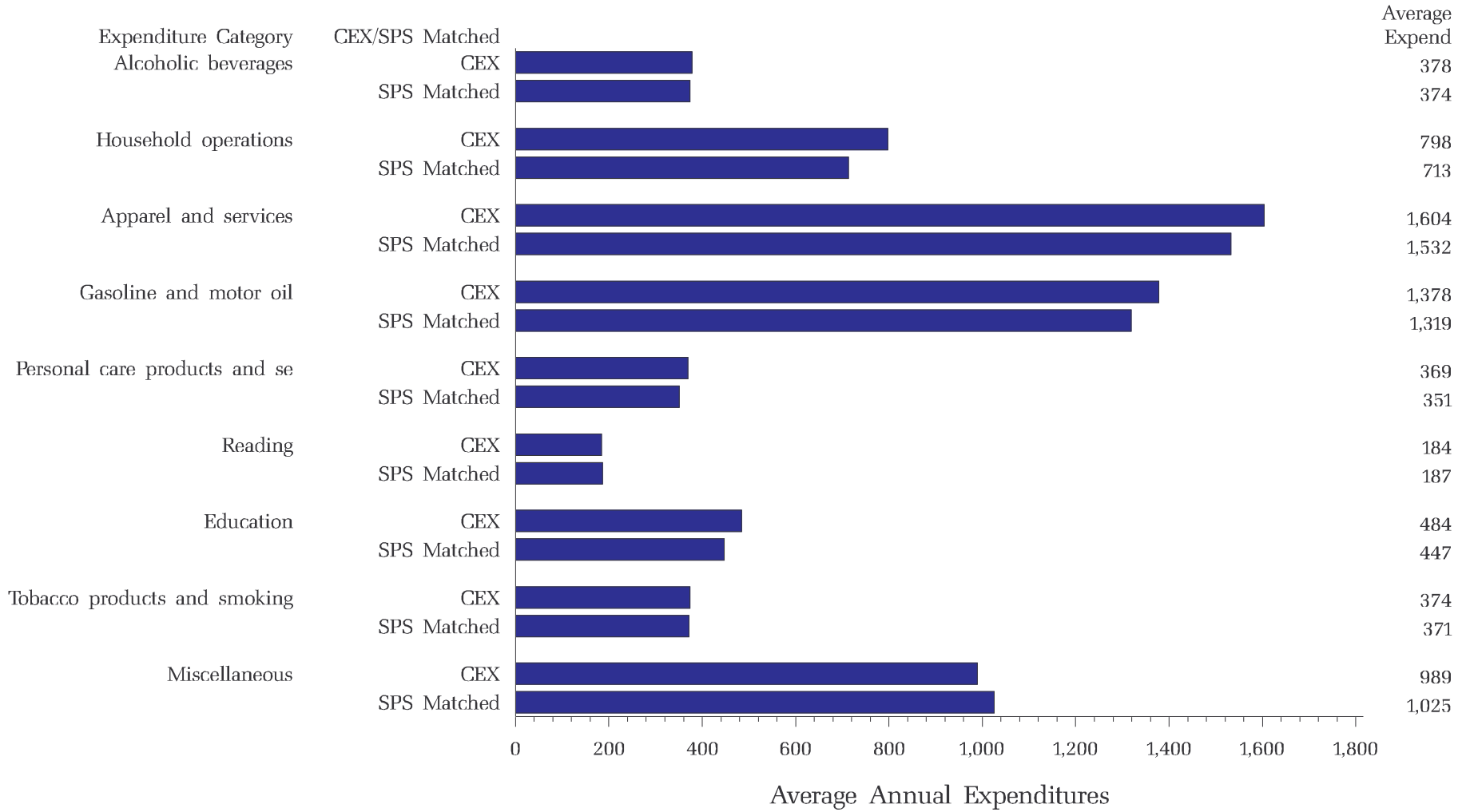
1999 HOUSEHOLD TOTAL INCOME = $40,000 to $50,000



Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

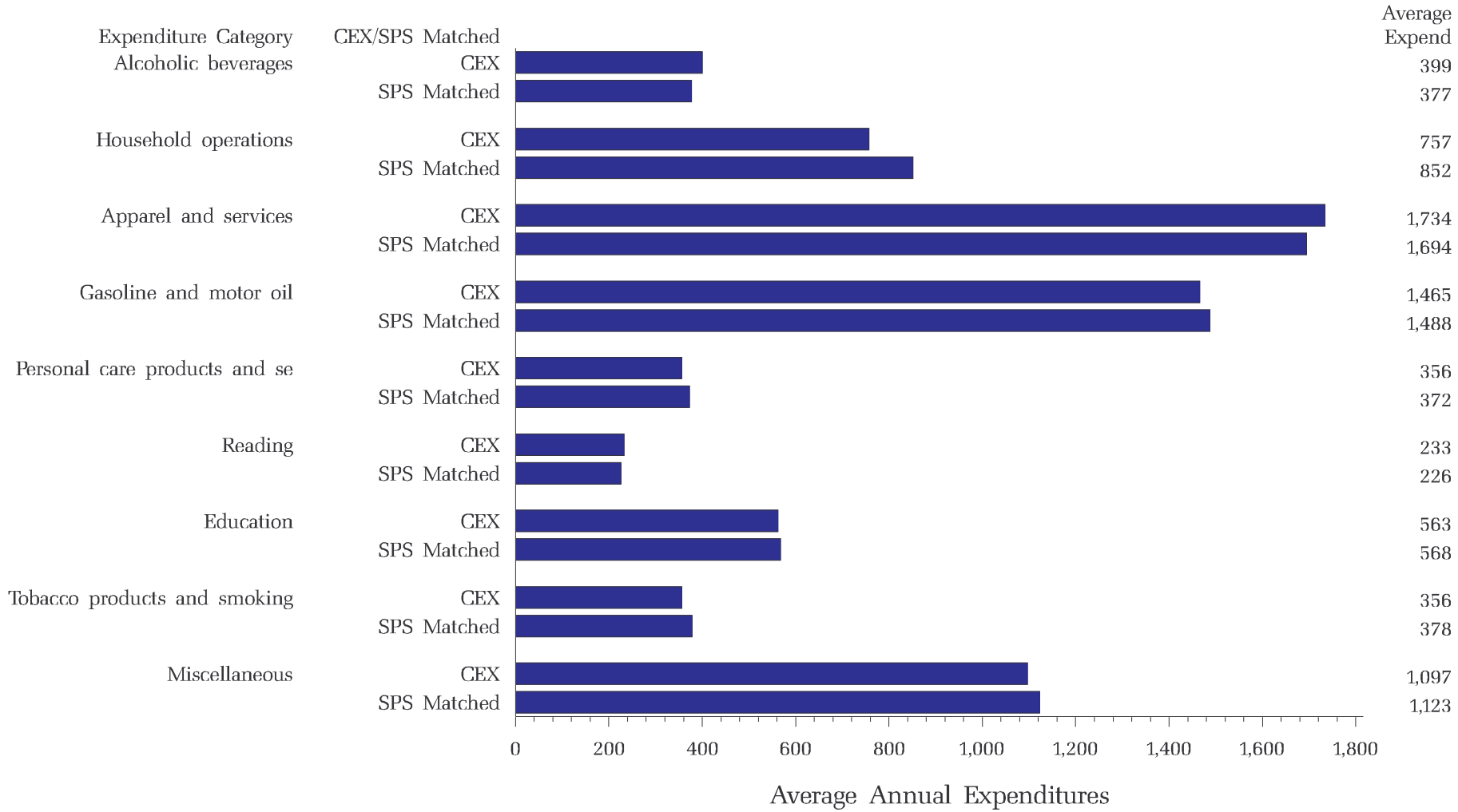
1999 HOUSEHOLD TOTAL INCOME = $50,000 to $60,000



Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

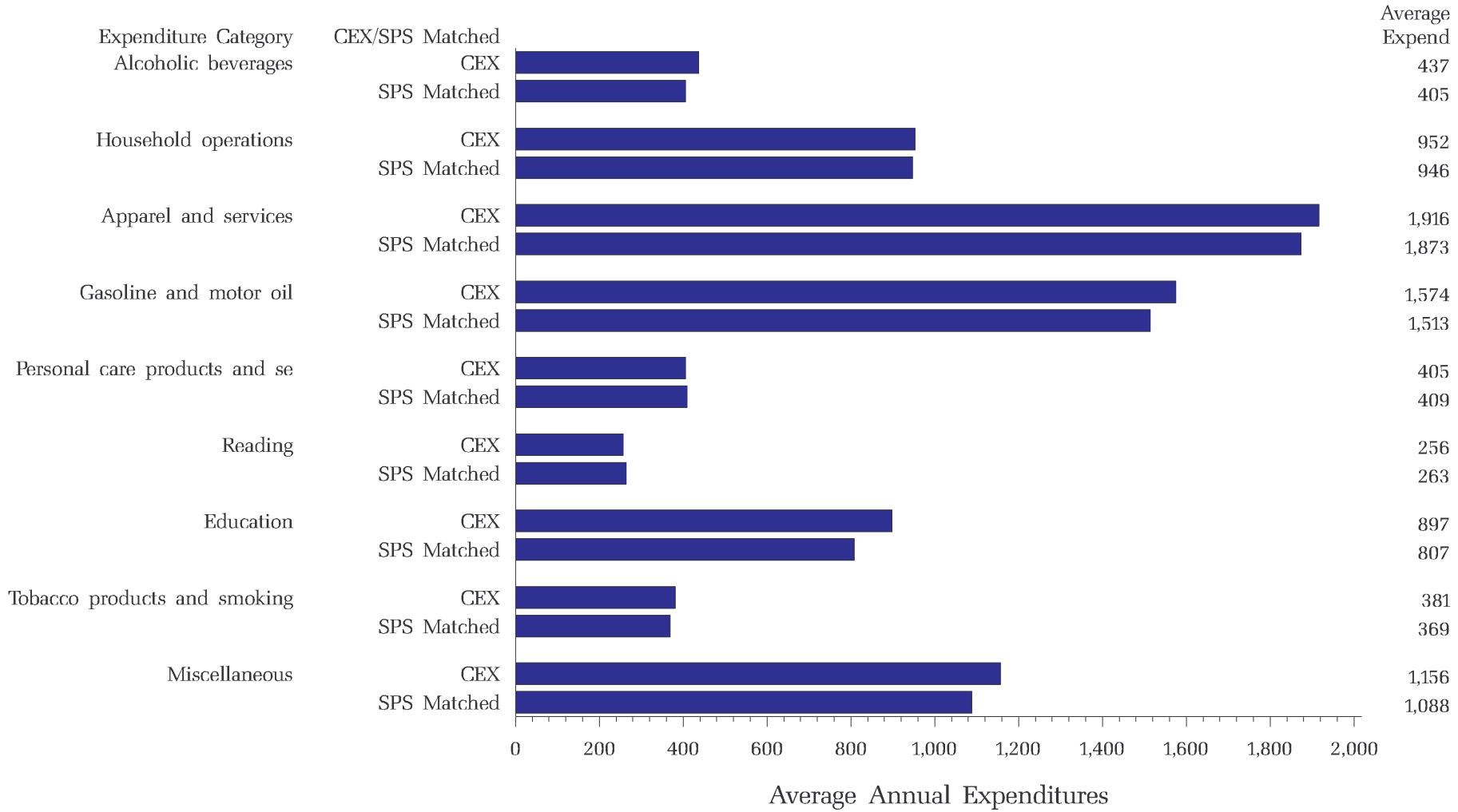
1999 HOUSEHOLD TOTAL INCOME = $60,000 to $70,000



Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

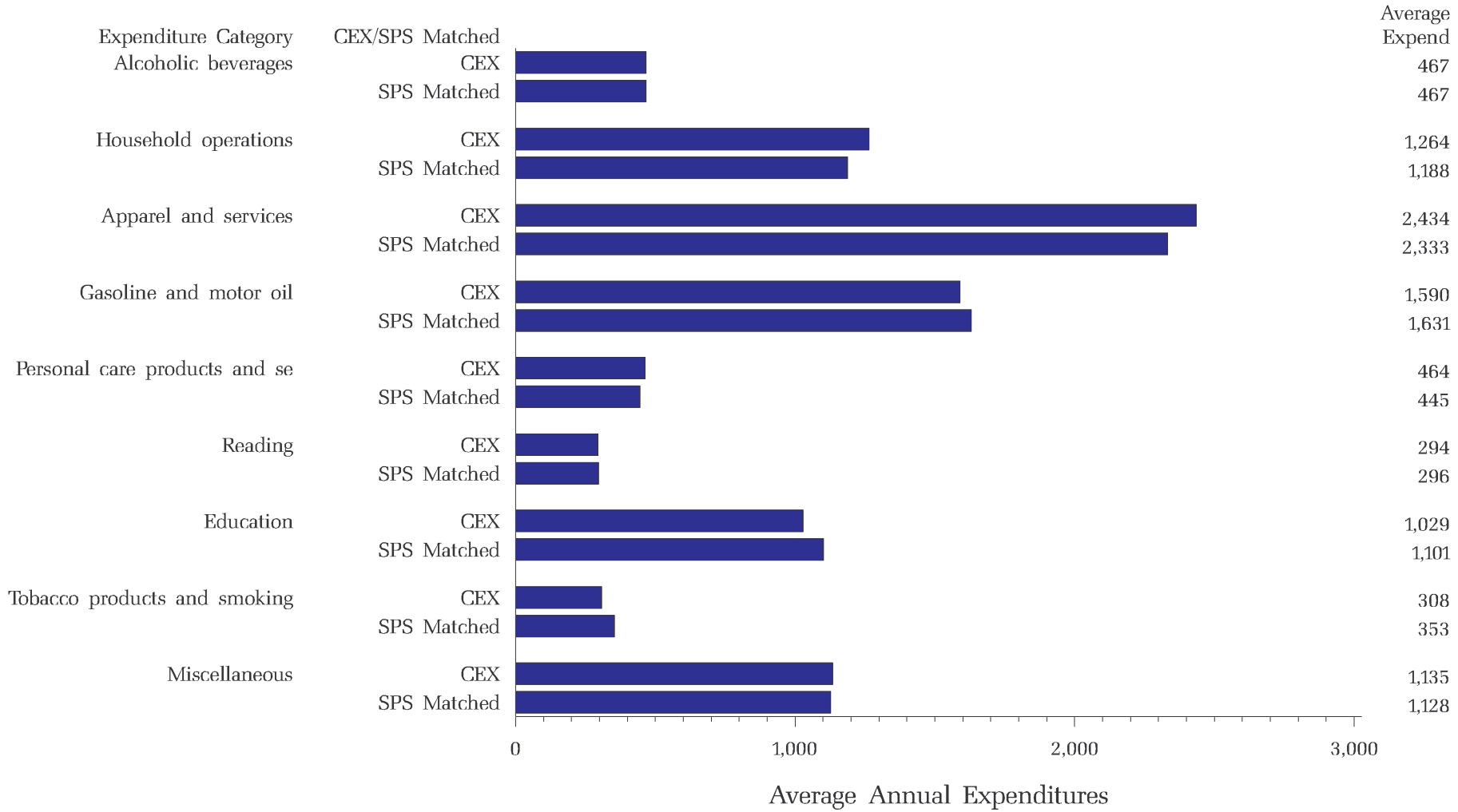
1999 HOUSEHOLD TOTAL INCOME = $70,000 to $80,000



Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

1999 HOUSEHOLD TOTAL INCOME=$80,000 to $100,000

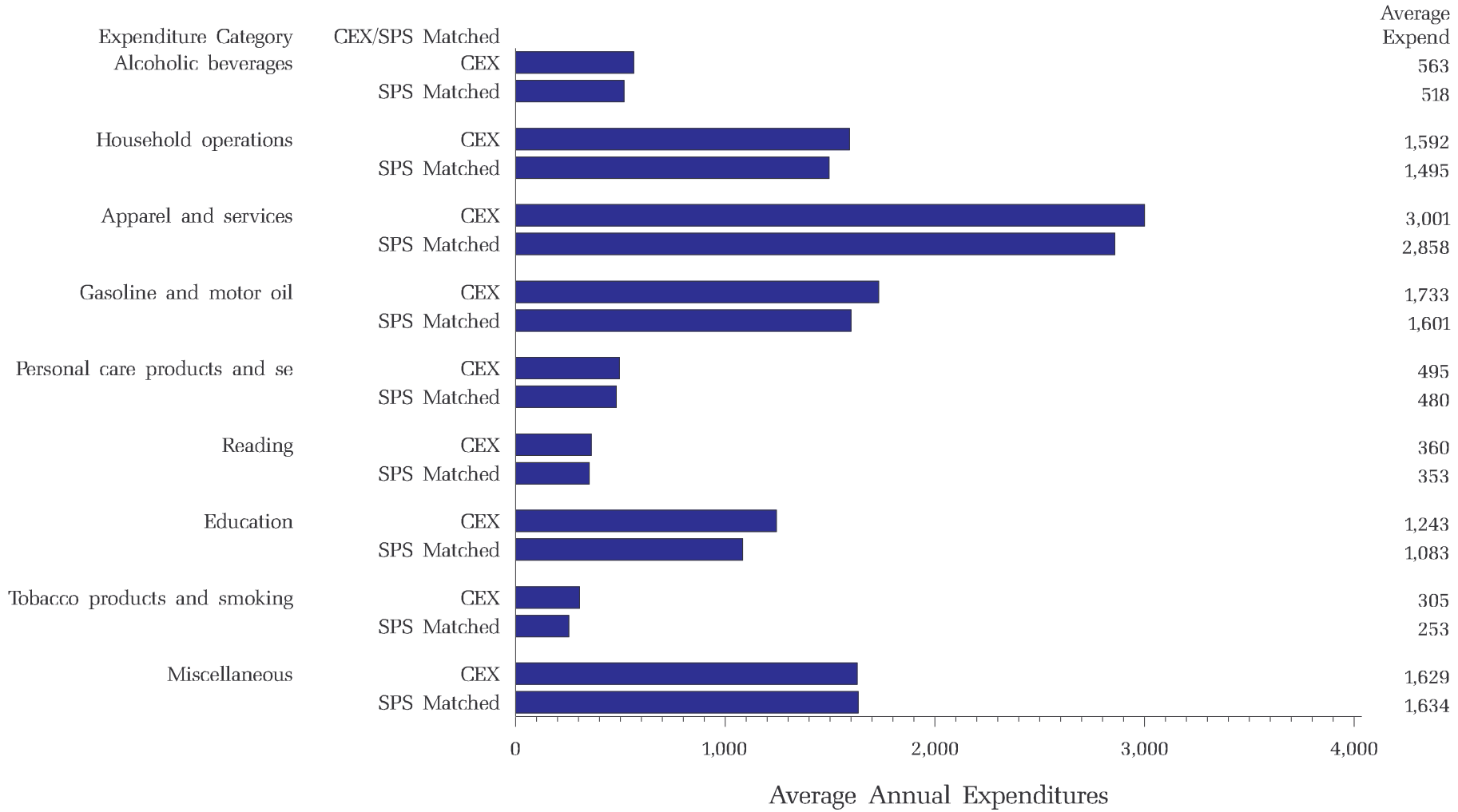


Appendix 4

Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

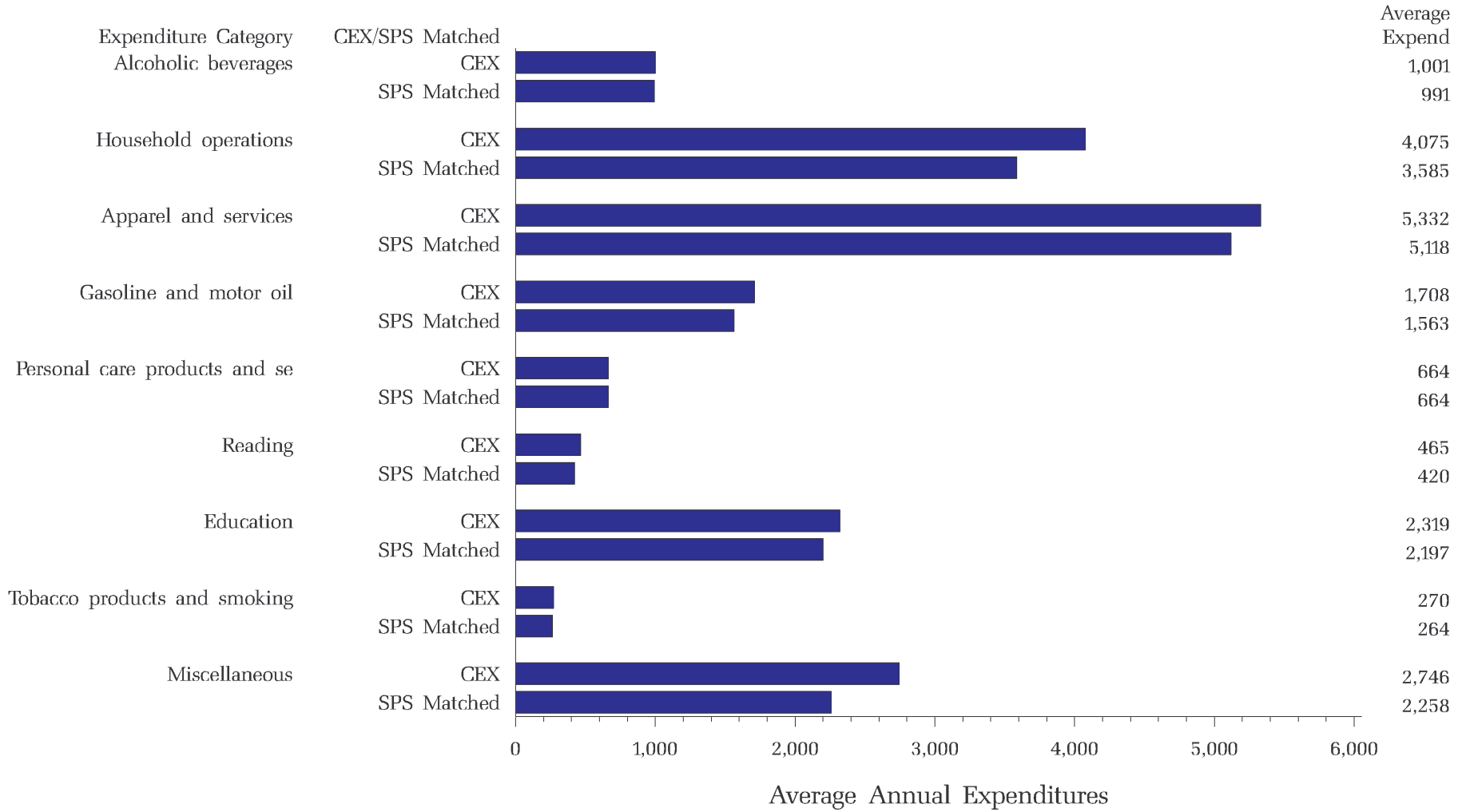
1999 HOUSEHOLD TOTAL INCOME=$100,000 to $130,000



Average Consumption for Selected Expenditure Categories

Average for CEX data compared to Average for the 7 SPS Matched Data Sets

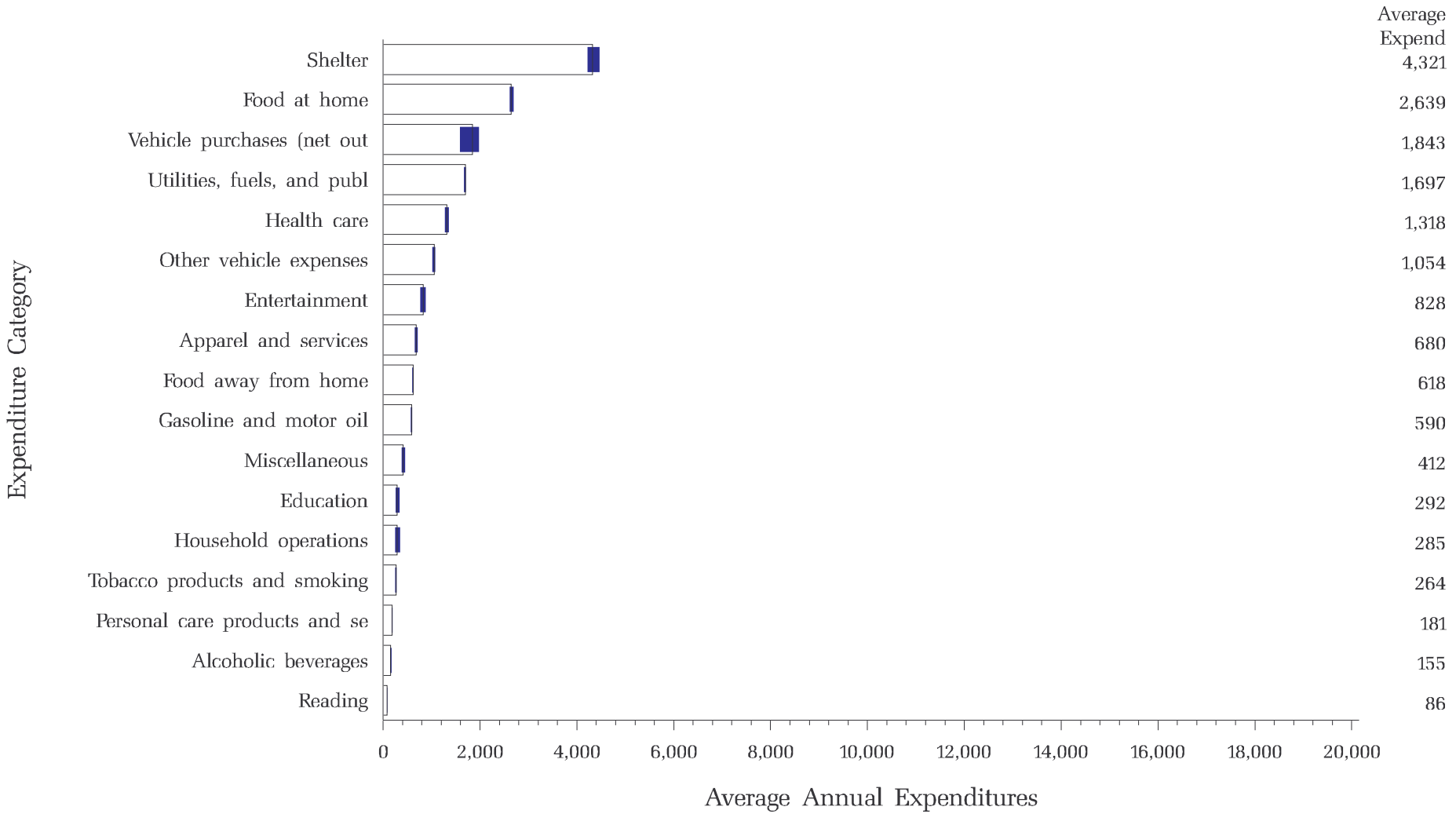
1999 HOUSEHOLD TOTAL INCOME = Over $130,000



Average Consumption for Selected Expenditure Categories

Average and Range for the 7 SPS Matched Data Sets

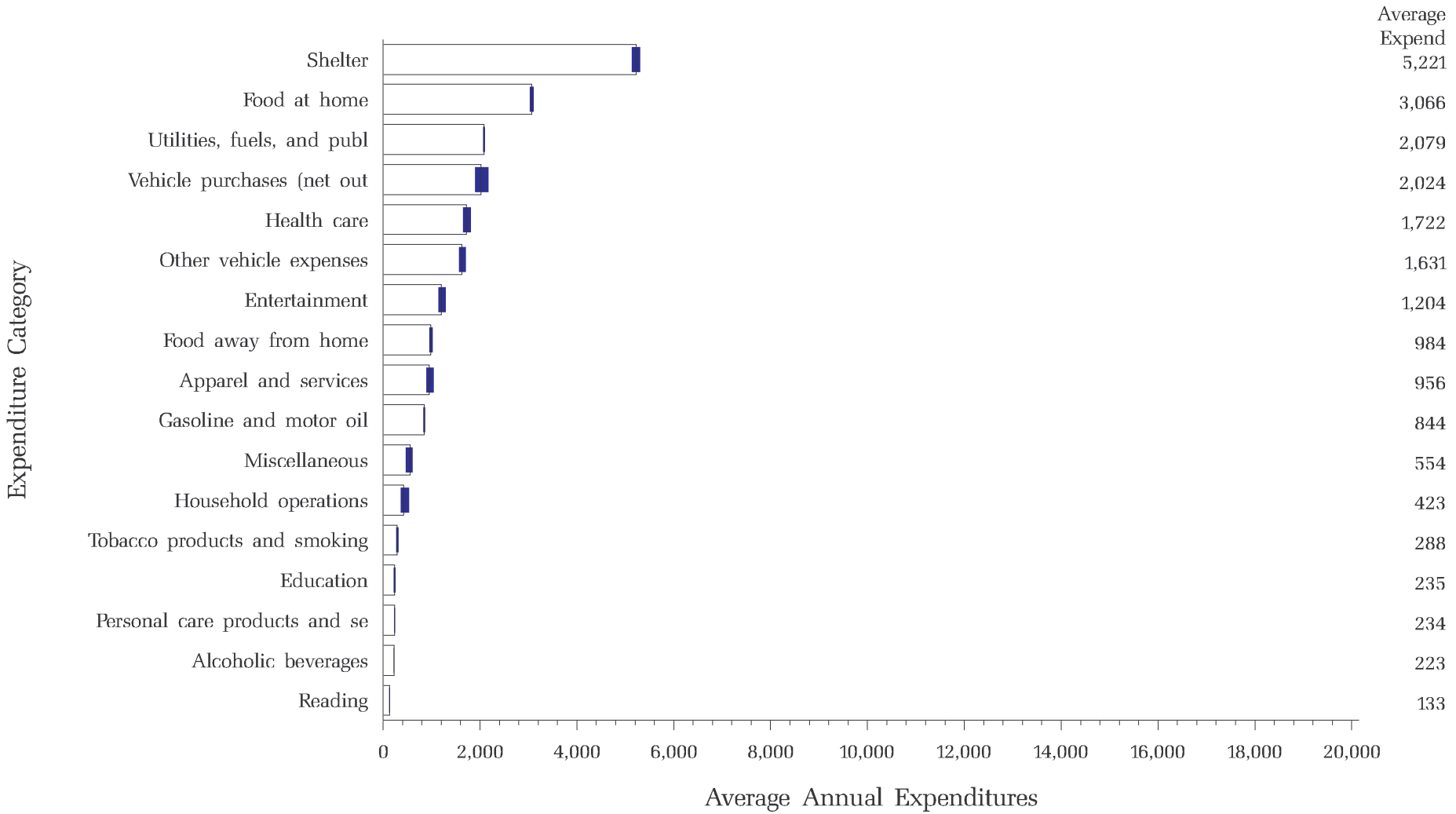
1999 HOUSEHOLD TOTAL INCOME=$0 to $20,000



Average Consumption for Selected Expenditure Categories

Average and Range for the 7 SPS Matched Data Sets

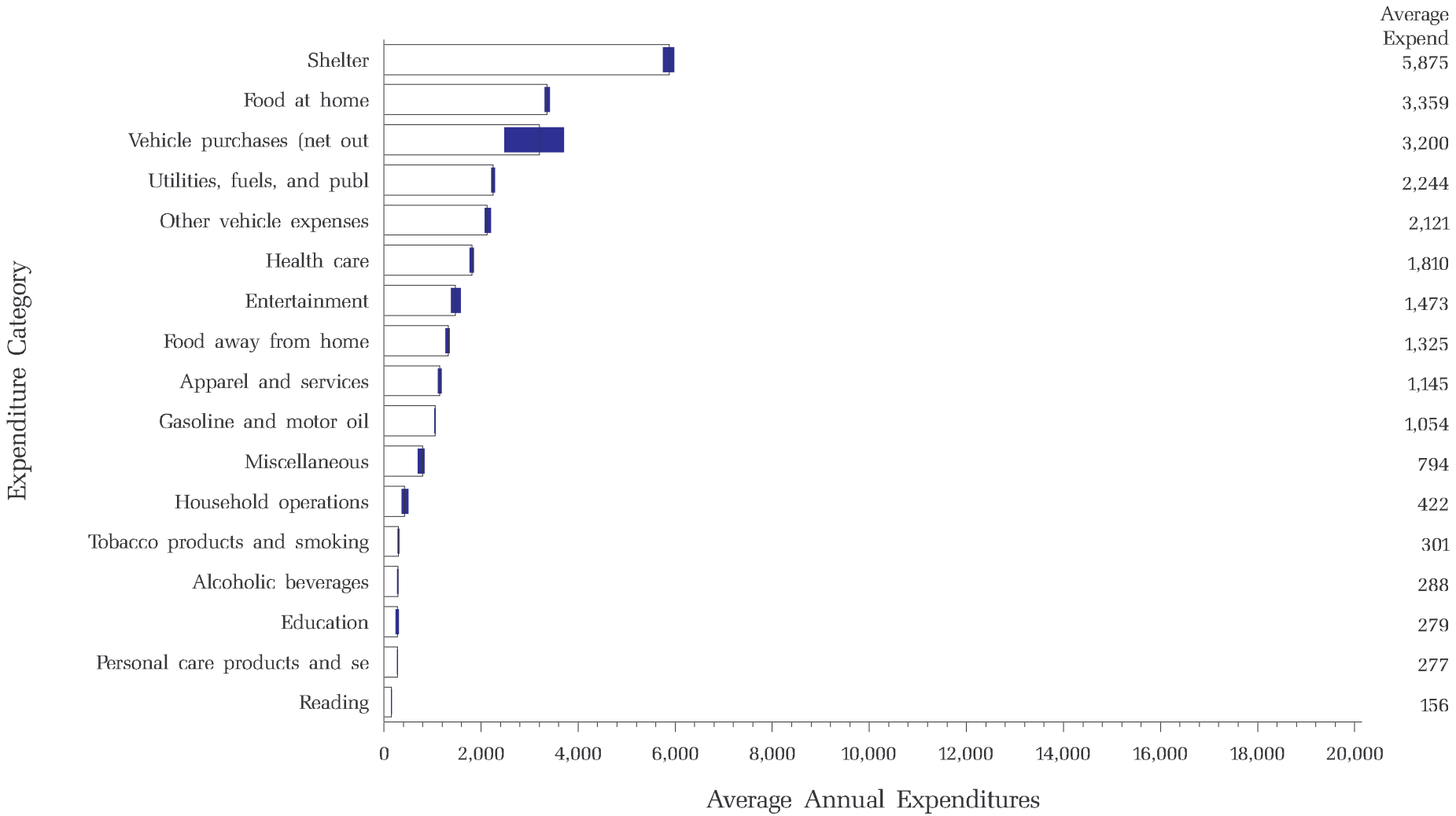
1999 HOUSEHOLD TOTAL INCOME = $20,000 to $30,000



Average Consumption for Selected Expenditure Categories

Average and Range for the 7 SPS Matched Data Sets

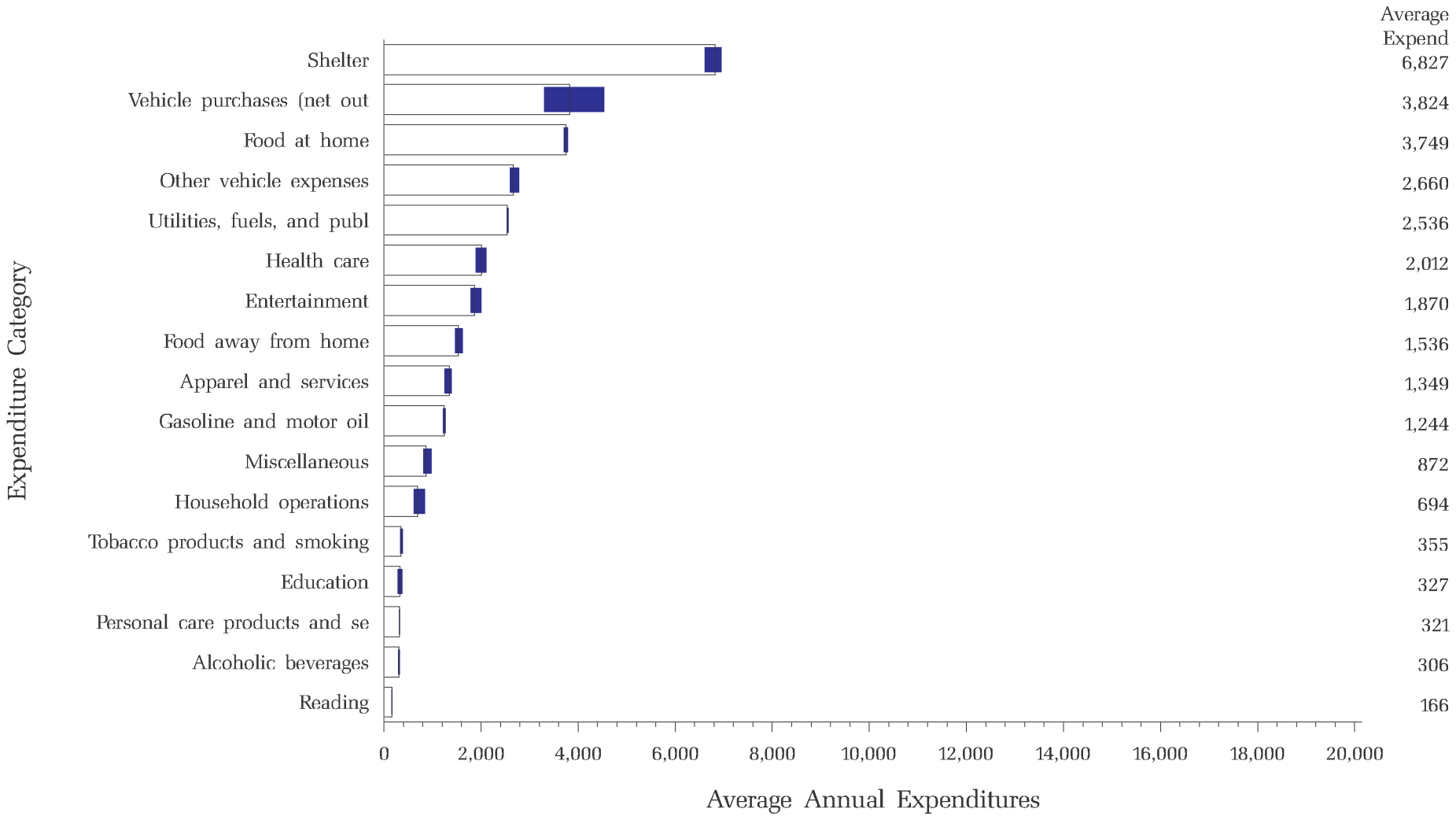
1999 HOUSEHOLD TOTAL INCOME = $30,000 to $40,000



Average Consumption for Selected Expenditure Categories

Average and Range for the 7 SPS Matched Data Sets

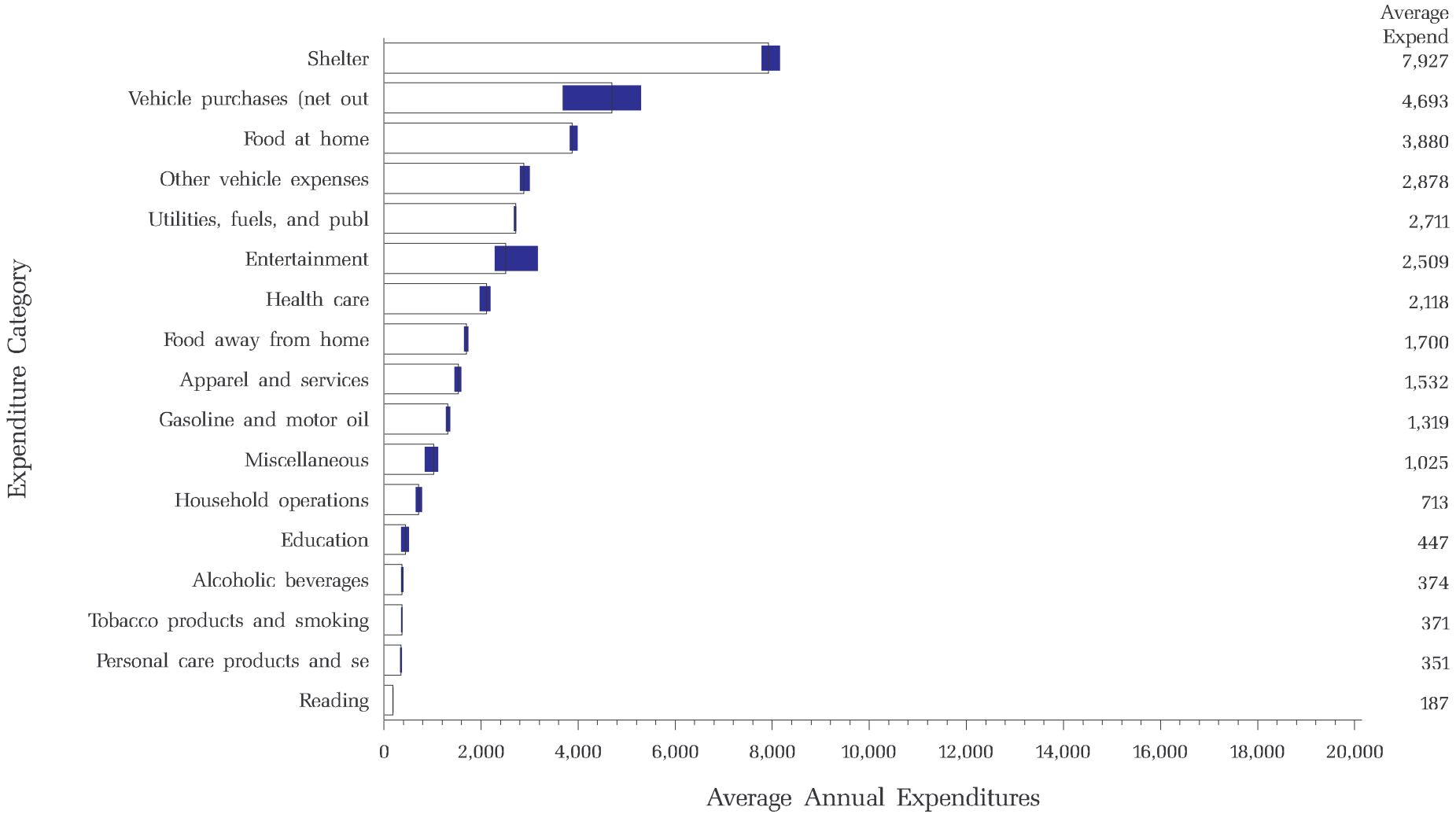
1999 HOUSEHOLD TOTAL INCOME = $40,000 to $50,000



Average Consumption for Selected Expenditure Categories

Average and Range for the 7 SPS Matched Data Sets

1999 HOUSEHOLD TOTAL INCOME = $50,000 to $60,000

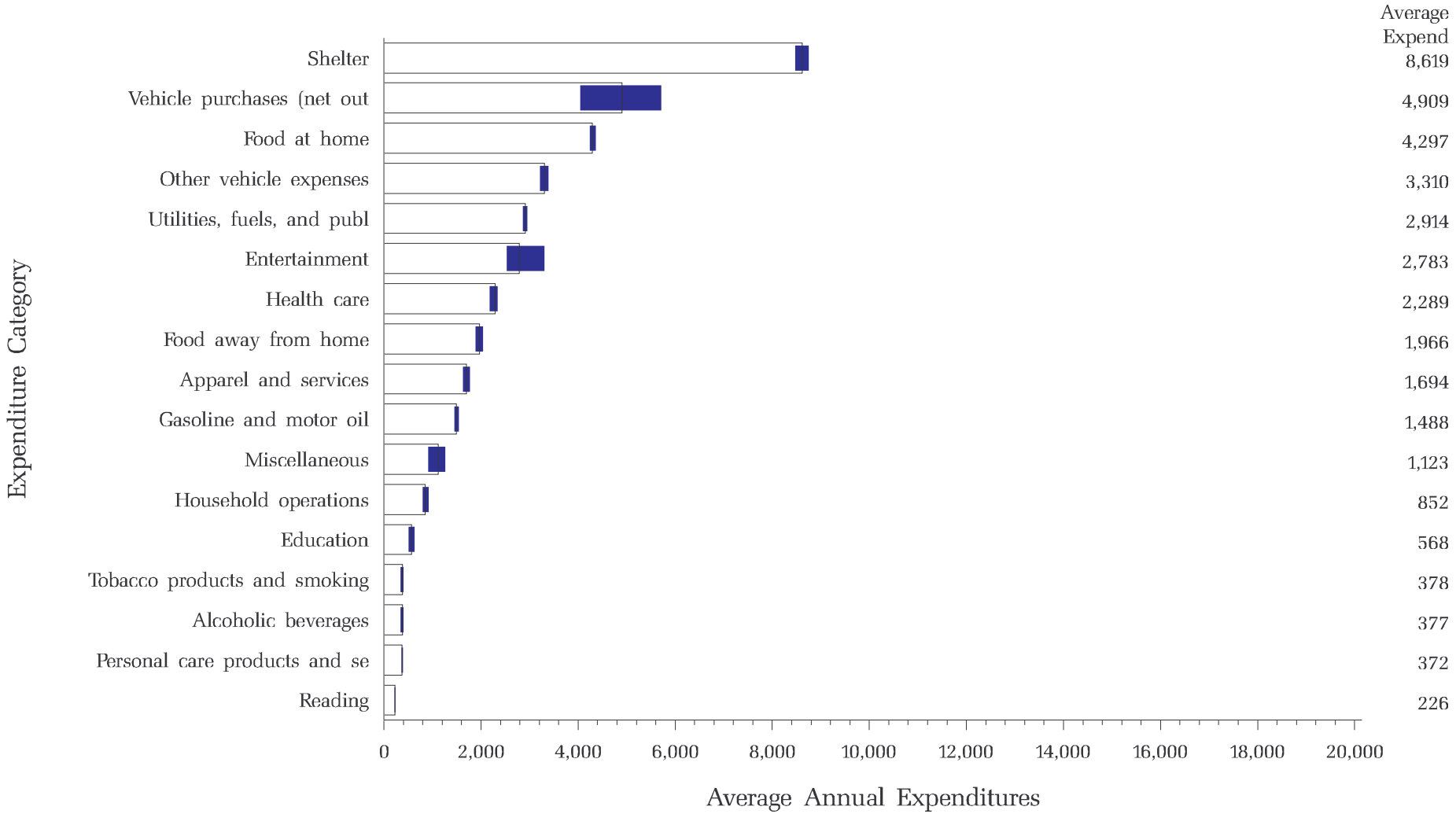


Appendix 5

Average Consumption for Selected Expenditure Categories

Average and Range for the 7 SPS Matched Data Sets

1999 HOUSEHOLD TOTAL INCOME = $60,000 to $70,000

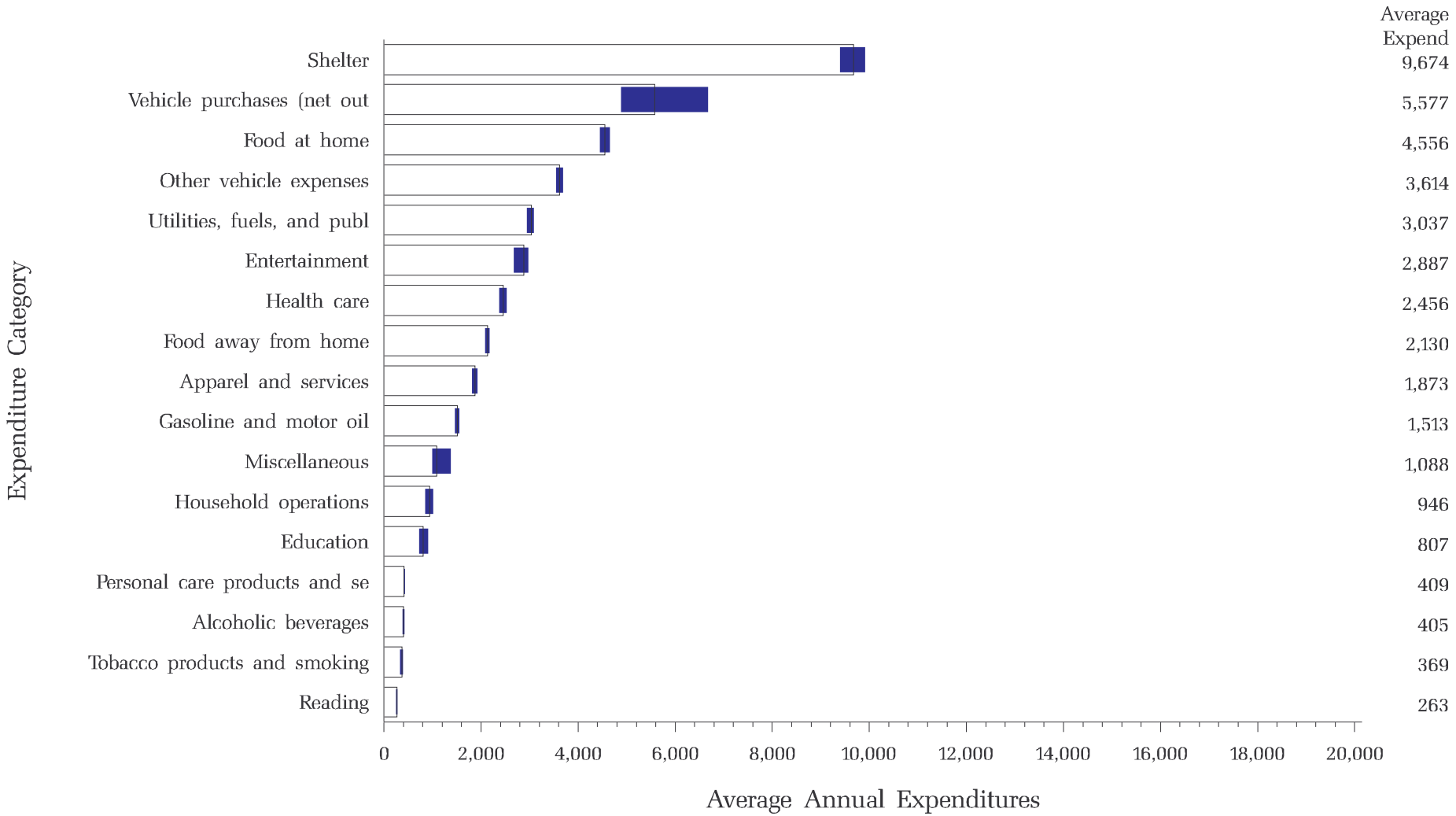


Appendix 5

Average Consumption for Selected Expenditure Categories

Average and Range for the 7 SPS Matched Data Sets

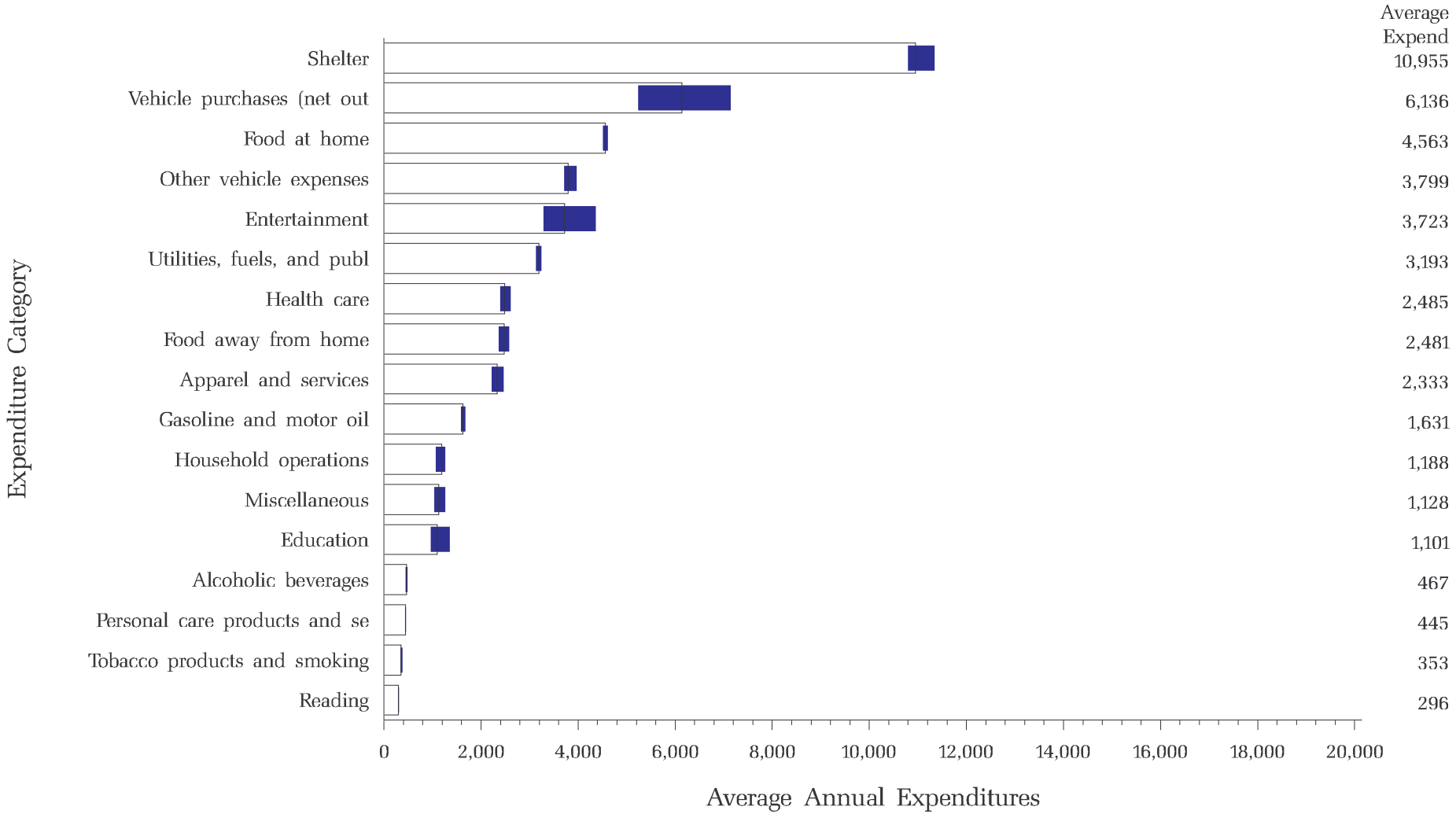
1999 HOUSEHOLD TOTAL INCOME = $70,000 to $80,000



Average Consumption for Selected Expenditure Categories

Average and Range for the 7 SPS Matched Data Sets

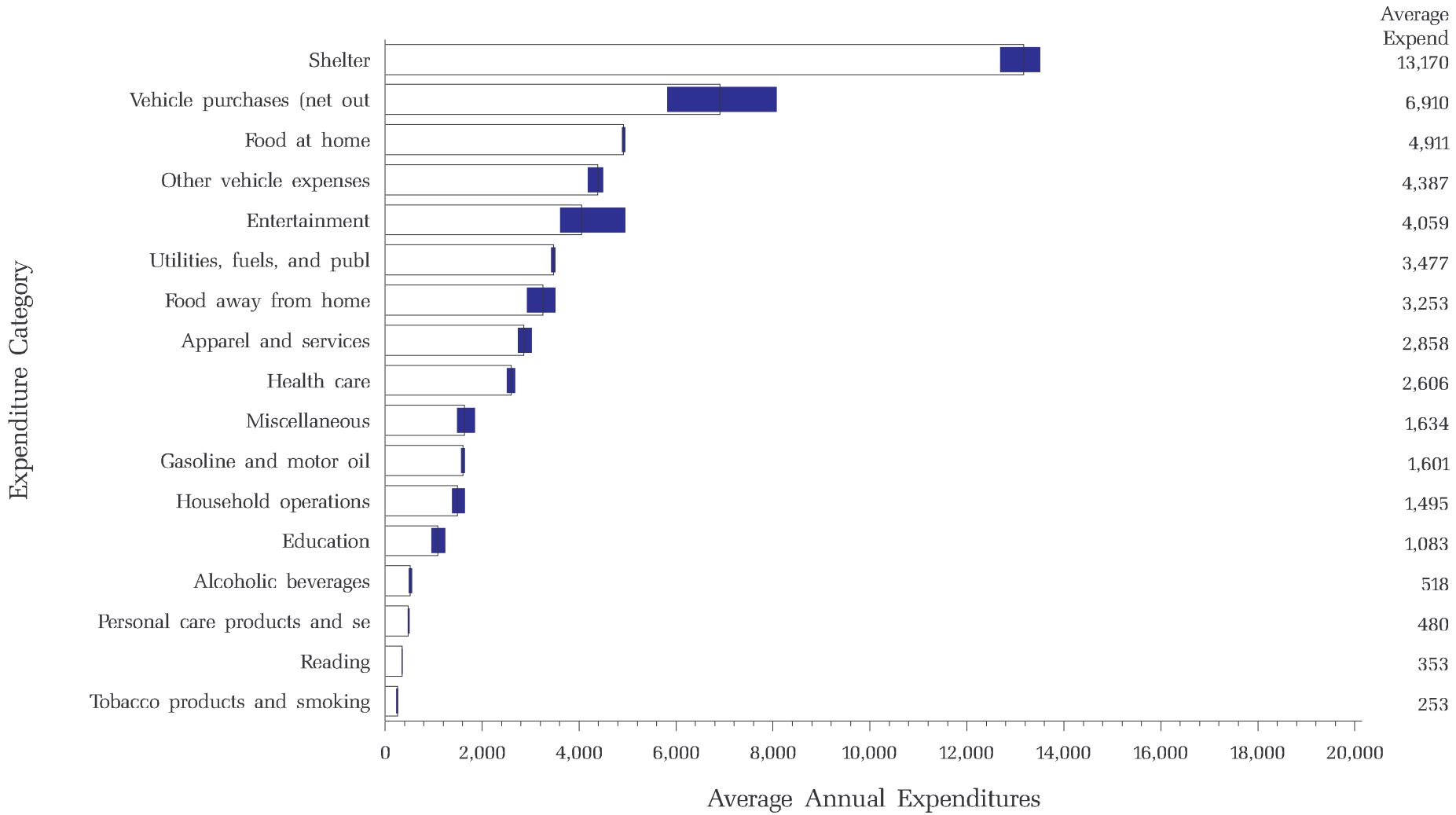
1999 HOUSEHOLD TOTAL INCOME=$80,000 to $100,000



Average Consumption for Selected Expenditure Categories

Average and Range for the 7 SPS Matched Data Sets

1999 HOUSEHOLD TOTAL INCOME=$100,000 to $130,000

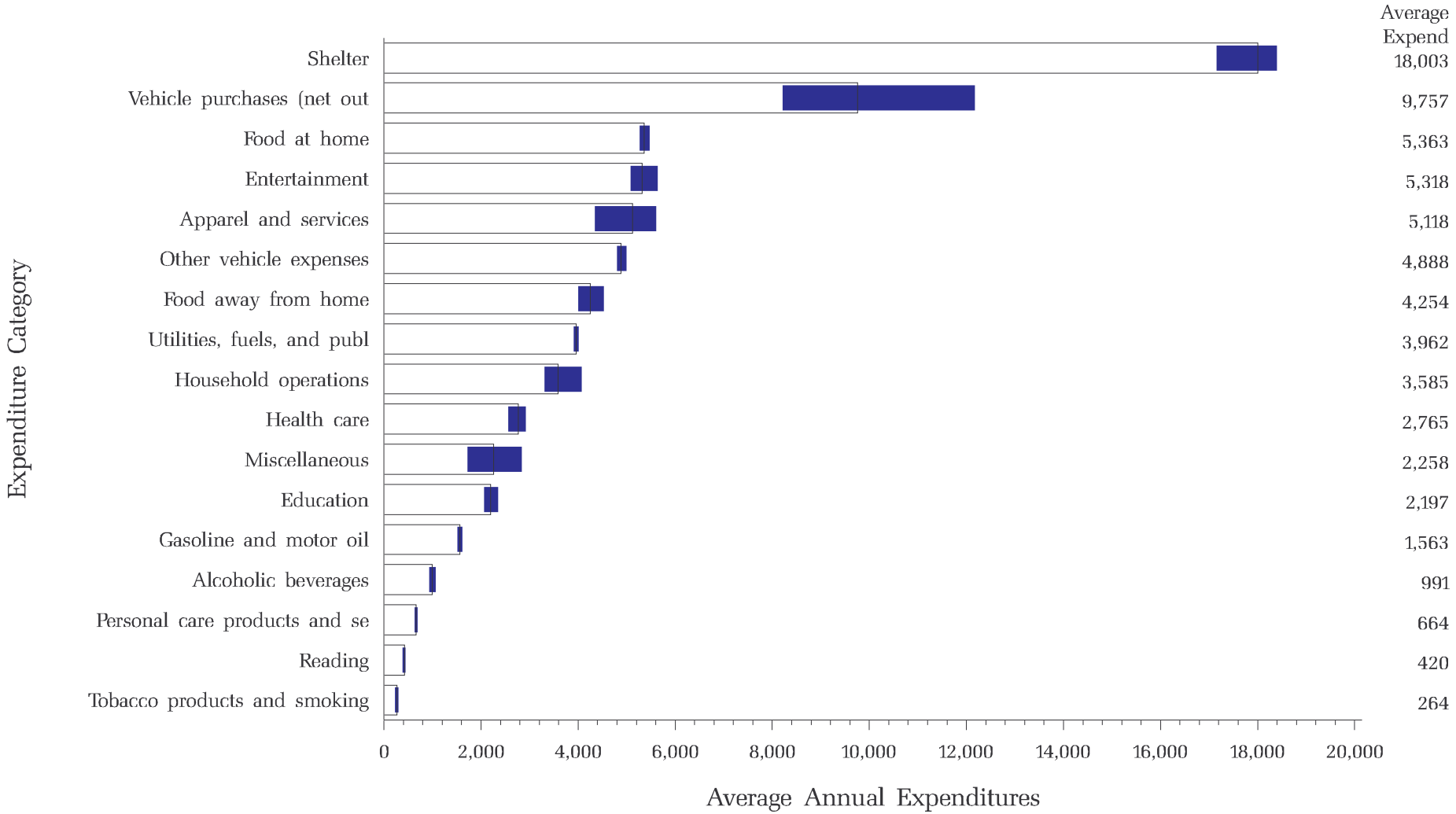


Appendix 5

Average Consumption for Selected Expenditure Categories

Average and Range for the 7 SPS Matched Data Sets

1999 HOUSEHOLD TOTAL INCOME = Over $130,000



Appendix 6

Taxable Items by UCC

Ucccode	ucccodelabel	surveycode	Current	Other	Baseadj
1000	STOCKS, BONDS, MUTUAL FUNDS	I	0	0	1.00
1010	SALE PRC STCK/BND/MUT FND, NET	I	0	0	1.00
1210	INVESTS TO FARM/BUSINESS	I	0	0	1.00
1220	ASSETS TAKEN FR FARM/BUSINESS	I	0	0	1.00
2010	CHANGE IN SAVINGS ACCOUNT	I	0	0	1.00
2020	CHANGE IN CHECKING ACCOUNT	I	0	0	1.00
2030	CHANGE IN AMT US SAVING BONDS	I	0	0	1.00
2120	OTHER NON-HEALTH INSURANCE	I	0	7	1.00
3000	CHANGE IN MONEY OWED TO CU	I	0	0	1.00
3100	SURRENDER OF INS POLICIES	I	0	0	1.00
6001	TOTAL AMT OWED 2ND INTERVIEW	I	0	0	1.00
6002	TOTAL AMT OWED 5TH INTERVIEW	I	0	0	1.00
6003	TOTAL AMT OWED 2ND INTER,CY Q1	I	0	0	1.00
6004	TOTAL AMT OWED 5TH INTER,CY Q1	I	0	0	1.00
6005	TOTAL AMT OWED 2ND INTER,CY+1,Q1	I	0	0	1.00
6006	TOTAL AMT OWED 5TH INTER,CY+1,Q1	I	0	0	1.00
10110	FLOUR	D	0	0	1.56
10120	PREPARED FLOUR MIXES	D	0	0	1.56
10210	CEREAL	D	0	0	1.56
10310	RICE	D	0	0	1.56
10320	PASTA CORNMEAL OTH CEREAL PRODS	D	0	0	1.56
20110	WHITE BREAD	D	0	0	1.47
20210	BREAD OTHER THAN WHITE	D	0	0	1.47
20310	FRESH BISCUITS, ROLLS, MUFFINS	D	0	0	1.47
20410	CAKES AND CUPCAKES	D	0	0	1.47
20510	COOKIES	D	0	0	1.47
20610	CRACKERS	D	0	0	1.47
20620	BREAD AND CRACKER PRODUCTS	D	0	0	1.47
20710	DOUGHNUTS,SWEETROLLS,COFFECAKE	D	0	0	1.47
20810	FROZEN & REFRIG. BAKERY PROD.	D	0	0	1.47
20820	FRESH PIES, TARTS, TURNOVERS	D	0	0	1.47
30110	GROUND BEEF EXCLUDE CANNED	D	0	0	1.23
30210	CHUCK ROAST	D	0	0	1.23
30310	ROUND ROAST	D	0	0	1.23
30410	OTHER ROAST	D	0	0	1.23
30510	ROUND STEAK	D	0	0	1.23
30610	SIRLOIN STEAK	D	0	0	1.23
30710	OTHER STEAK	D	0	0	1.23
30810	OTHER BEEF (EXCLUDE CANNED)	D	0	0	1.23
40110	BACON	D	0	0	1.23
40210	PORK CHOPS	D	0	0	1.23
40310	HAM (EXCLUDE CANNED)	D	0	0	1.23
40410	OTHER PORK	D	0	0	1.23
40510	PORK SAUSAGE	D	0	0	1.23
40610	CANNED HAM	D	0	0	1.23
50110	FRANKFURTERS	D	0	0	1.23
50210	BOLOGNA, LIVERWURST, SALAMI	D	0	0	1.23

Taxable Items by UCC

50310	OTHER LUNCHMEAT	D	0	0	1.23
50410	LAMB AND ORGAN MEATS	D	0	0	1.23
50900	MUTTON, GOAT, GAME	D	0	0	1.23
60110	FRESH & FROZEN WHOLE CHICKEN	D	0	0	1.23
60210	FRESH OR FROZEN CHICKEN PARTS	D	0	0	1.23
60310	OTHER POULTRY	D	0	0	1.23
70110	CANNED FISH AND SEAFOOD	D	0	0	0.81
70230	FRESH FISH & SHELLFISH	D	0	0	0.81
70240	FROZEN FISH & SHELLFISH	D	0	0	0.81
80110	EGGS	D	0	0	0.93
90110	FRESH MILK ALL TYPES	D	0	0	0.83
90210	CREAM	D	0	0	1.28
100110	BUTTER	D	0	0	1.28
100210	CHEESE	D	0	0	1.28
100410	ICE CREAM AND RELATED PRODUCTS	D	0	0	1.28
100510	OTHER DAIRY PRODUCTS	D	0	0	1.28
110110	APPLES	D	0	0	1.11
110210	BANANAS	D	0	0	1.11
110310	ORANGES	D	0	0	1.11
110410	OTHER FRESH FRUITS	D	0	0	1.11
110510	CITRUS FRUITS EXCL. ORANGES	D	0	0	1.11
120110	POTATOES	D	0	0	1.11
120210	LETTUCE	D	0	0	1.11
120310	TOMATOES	D	0	0	1.11
120410	OTHER FRESH VEGETABLES	D	0	0	1.11
130110	FROZEN ORANGE JUICE	D	0	0	1.61
130121	FROZEN FRUITS	D	0	0	1.61
130122	FROZEN FRUIT JUICES	D	0	0	1.61
130211	FRESH FRUIT JUICE	D	0	0	1.61
130212	CANNED/BOTTLE FRUIT JUICE	D	0	0	1.61
130310	CANNED FRUITS	D	0	0	1.61
130320	DRIED FRUITS	D	0	0	1.61
140110	FROZEN VEGETABLES	D	0	0	1.61
140210	CANNED BEANS	D	0	0	1.61
140220	CANNED CORN	D	0	0	1.61
140230	CANNED VEGETABLES MISC	D	0	0	1.61
140310	OTHER PROCESSED VEGETABLES	D	0	0	1.61
140320	OTHER PEAS	D	0	0	1.61
140330	OTHER BEANS	D	0	0	1.61
140340	OTHER VEGETABLES MISC	D	0	0	1.61
140410	FROZEN VEGETABLE JUICES	D	0	0	1.61
140420	FRESH & CANNED VEGETABLE JUICES	D	0	0	1.61
150110	CANDY AND CHEWING GUM	D	0	0	2.70
150211	SUGAR	D	0	0	2.70
150212	ARTIFICIAL SWEETENERS	D	0	0	2.70
150310	OTHER SWEETS	D	0	0	2.70
160110	MARGARINE	D	0	0	1.28
160211	FATS & OILS	D	0	0	1.28

Taxable Items by UCC

160212	SALAD DRESSINGS	D	0	0	1.28
160310	NON-DIARY CREAM SUBSTITUTES	D	0	0	1.28
160320	PEANUT BUTTER	D	0	0	1.28
170110	COLA DRINKS	D	1	0	1.96
170210	OTHER CARBONATED DRINKS	D	1	0	1.96
170310	ROASTED COFFEE	D	0	0	1.96
170410	INSTANT/FREEZE DRIED COFFEE	D	0	0	1.96
170510	NONCARB FRUT FLAV/LEMADE NONFROZ	D	0	0	1.96
170520	TEA	D	0	0	1.96
170530	OTHER NONCARB. BEVERAGES/ICE	D	0	0	1.96
180110	SOUP	D	0	0	1.20
180210	FROZEN MEALS	D	0	0	1.20
180220	FROZ/PREP. FOOD OTH THAN MEALS	D	0	0	1.20
180310	POTATO CHIPS AND OTHER SNACKS	D	0	0	1.20
180320	NUTS	D	0	0	1.20
180410	SALT/OTHER SEASONINGS & SPICES	D	0	0	1.20
180420	OLIVES, PICKLES, RELISHES	D	0	0	1.20
180510	SAUCES AND GRAVIES	D	0	0	1.20
180520	OTHER CONDIMENTS	D	0	0	1.20
180611	PREPARED SALADS	D	1	0	1.20
180612	PREPARED DESSERTS	D	1	0	1.20
180620	BABY FOOD	D	0	0	1.20
180710	MISC. PREPARED FOODS	D	1	0	1.20
180720	VITAMIN SUPPLEMENT	D	1	0	1.20
190111	LUNCH AT FAST FOOD	D	1	0	1.35
190112	LUNCH AT FULL SERRVICE	D	1	0	1.35
190113	LUNCH AT VENDING MACHINE	D	0.57	0	1.35
190114	LUNCH AT EMPLOYER	D	1	0	1.35
190115	LUNCH AT BOARD	D	0	0	1.35
190116	LUNCH AT CATERED AFFAIRS	D	1	0	1.35
190211	DINNER AT FAST FOOD	D	1	0	1.35
190212	DINNER AT FULL SERVICE	D	1	0	1.35
190213	DINNER AT VENDING MACHINE	D	0.57	0	1.35
190214	DINNER AT EMPLOYER	D	1	0	1.35
190215	DINNER AT BOARD	D	0	0	1.35
190216	DINNER AT CATERED AFFAIRS	D	1	0	1.35
190311	SNACKS AT FAST FOOD	D	1	0	1.35
190312	SNACKS AT FULL SERVICE	D	1	0	1.35
190313	SNACKS AT VEND MACHINE	D	0.57	0	1.35
190314	SNACKS AT EMPLOYER	D	1	0	1.35
190315	SNACKS AT BOARD	D	0	0	1.35
190316	SNACKS AT CATERED AFFAIRS	D	1	0	1.35
190321	BREAKFAST AT FAST FOOD	D	1	0	1.35
190322	BREAKFAST AT FULL SERVICE	D	1	0	1.35
190323	BREAKFAST AT VENDING MACHINE	D	1	0	1.35
190324	BREAKFAST AT EMPLOYER	D	1	0	1.35
190325	BREAKFAST AT BOARD	D	1	0	1.35
190326	BREAKFAST AT CATERED AFFAIRS	D	1	0	1.35

Taxable Items by UCC

190901	BOARD (INCLUD AT SCHOOL)	I	0	0	1.35
190902	CATERED AFFAIRS	I	1	0	1.35
190903	FOOD OUT OF TOWN TRIPS	I	0.9	0	1.35
190904	FOOD PREPARED BY CU ON TRIPS	I	0	0	1.00
190911	BOARD AT FAST FOOD	D	1	0	1.35
190912	BOARD AT FULL SERVICE	D	1	0	1.35
190913	BOARD AT VENDING MACHINE	D	0.57	0	1.35
190914	BOARD AT EMPLOYER	D	1	0	1.35
190915	BOARD AT BOARD	D	0	0	1.35
190916	BOARD AT CATERED AFFAIRS	D	1	0	1.35
190921	CATERED AFF AT FAST FOOD	D	1	0	1.35
190922	CATERED AFF AT FULL SERVICE	D	1	0	1.35
190923	CATERED AFF AT VEND MACHINE	D	0.57	0	1.35
190924	CATERED AFF AT EMPLOYER	D	1	0	1.35
190925	CATERED AFF AT BOARD	D	0	0	1.35
190926	CATERED AFF AT CATERED AFF	D	1	0	1.35
200111	BEER AND ALE AT HOME	D	1	1	2.12
200112	NON ALCOHOLIC BEER	D	0	0	2.12
200210	WHISKEY AT HOME	D	0	3	2.74
200310	WINE AT HOME	D	1	2	1.93
200410	OTHER ALCOHOLIC BEV. AT HOME	D	0	3	2.74
200511	BEER AT FAST FOOD	D	1	4	2.12
200512	BEER AT FULL SERVICE	D	1	4	2.12
200513	BEER AT VENDING MACHINE	D	1	4	2.12
200514	EMPTY BEER AT EMPLOYER	D	1	4	2.12
200515	EMPTY BEER AT BOARD	D	1	4	2.12
200516	BEER AT CATERED AFFAIRS	D	1	4	2.12
200521	WINE AT FAST FOOD	D	1	5	1.93
200522	WINE AT FULL SERVICE	D	1	5	1.93
200523	WINE AT VENDING MACHINE	D	1	5	1.93
200524	EMPTY WINE AT EMPLOYER	D	1	5	1.93
200525	EMPTY WINE AT BOARD	D	1	5	1.93
200526	WINE AT CATERED AFFAIRS	D	1	5	1.93
200531	ALC. BEV EXC BEER/WINE FAST FD	D	1	6	2.74
200532	ALC. BEV EXC B/W FULL SERV	D	1	6	2.74
200533	ALC. BEV B/W VEND MACH	D	1	6	2.74
200534	EMPTY ALC BEV EXC B/W AT EMP	D	1	6	2.74
200535	EMPTY ALC BEV EXC B/W AT BOARD	D	1	6	2.74
200536	OTH ALC. BEV AWAY FROM HOME	D	1	6	2.74
200900	ALC. BEV. PURCHASED ON TRIPS	I	0.725	6	2.74
210110	RENT OF DWELLING	I	0	19	0.93
210210	LODGING AWAY FROM HOME	I	0.725	0	1.00
210310	HOUSING FOR SOMEONE AT SCHOOL	I	0	0	1.00
210901	GROUND RENT OWND	I	0	19	1.00
210902	GROUND RENT OWNV	I	0	19	1.00
220111	FIRE/EXTENDED COVERAGE OWND	I	0	7	1.12
220112	FIRE/EXTENDED COVERAGE OWNV	I	0	7	1.12
220121	HOMEOWNERS INSURANCE OWND	I	0	7	1.12

Taxable Items by UCC

220122	HOMEOWNERS INSURANCE OWNV	I	0	7	1.12
220211	PROPERTY TAXES OWND	I	0	0	1.00
220212	PROPERTY TAXES OWNV	I	0	0	1.00
220311	MORTGAGE INTEREST OWND	I	0	0	1.00
220312	MORTGAGE INTEREST OWNV	I	0	0	1.00
220313	INTEREST, LMP SUM HM EQ LN, OWND	I	0	0	1.00
220314	INTEREST, LMP SUM HM EQ LN, OWNV	I	0	0	1.00
220321	PREPAYMENT PENALTY OWND	I	0	0	1.00
220322	PREPAYMENT PENALTY OWNV	I	0	0	1.00
220400	PURCHASE OF PROPERTY	D	0	0	1.00
220410	HOME PURCHASE	D	0	0	1.00
220511	W/W CARPET NOT INST ORIG OWND	I	1	0	1.69
220512	CAPITAL IMPROVE MATERIALS OWND	I	1	0	1.12
220513	CAPITAL IMPROVE MATERIALS OWNV	I	1	0	1.12
220611	CAP IMPROVE LABOR/MAT OWND	I	1	0	1.12
220612	DWASH/DISP/HOOD CAP IMP	I	1	0	1.79
220614	W/W CARPET INST ORIG OWND	I	1	0	1.69
220615	CAP IMPROVE LABOR/MAT OWNV	I	1	0	1.12
220616	W/W CARPET ORIG OWND	I	1	0	1.69
220901	PARKING OWND	I	0	0	1.72
220902	PARKING OWNV	I	0	0	1.72
230112	PAINTING/PAPERING LABOR/MAT OWND	I	1	0	1.12
230113	PLMB/WTR HEAT LABOR/MAT OWND	I	1	0	1.12
230114	HEAT/AC/ELEC LABOR/MAT OWND	I	1	0	1.12
230115	ROOFING/GUTTERS LABOR/MAT OWND	I	1	0	1.12
230117	REPL DISHWASH/DISP/HOOD RNTR	I	1	0	1.79
230118	REPL DISHWASH/DISP/HOOD OWND	I	1	0	1.79
230121	HRD SURFACE FLOOR LABOR/MAT RNTR	I	1	0	1.69
230122	HRD SURFACE FLOOR LABOR/MAT OWND	I	1	0	1.69
230123	HRD SURFACE FLOOR LABOR/MAT OWNV	I	1	0	1.69
230131	W/W CARPET INST RNTR	I	1	0	1.69
230132	W/W CARPET INST REPL OWND	I	1	0	1.69
230133	W/W CARPET REPL OWND	I	1	0	1.69
230134	W/W CARPET ORIG RNTR	I	1	0	1.69
230141	REPAIR-DISPL/DWSHR/RANG HD RNTR	I	1	0	1.79
230142	REPAIR-DISPL/DWSHR/RANG HD OWND	I	1	0	1.79
230150	REP/MAINT LABOR/MAT RNTR	I	1	0	1.12
230151	OTH REP/MAINT LABOR/MAT OWND	I	1	0	1.12
230152	OTH REP/MAINT LABOR/MAT OWNV	I	1	0	1.12
230901	PROP MANAGEMENT OWND	I	0	0	1.00
230902	PROP MANAGEMENT OWNV	I	0	0	1.00
240111	PAINT/WALLPAPER AND SUPP RNTR	I	1	0	1.12
240112	PAINT/WALLPAPER AND SUPP OWND	I	1	0	1.12
240113	PAINT/WALLPAPER AND SUPP OWNV	I	1	0	1.12
240121	EQUIP FOR PAINT/WPAPER RNTR	I	1	0	1.12
240122	EQUIP FOR PAINT/WPAPER OWND	I	1	0	1.12
240123	EQUIP FOR PAINT/WPAPER OWNV	I	1	0	1.12
240211	MAT FOR PANL/ROOF/SIDING,ETC RNTR	I	1	0	1.12

Taxable Items by UCC

240212	MAT FOR PANEL/SIDING, ETC OWND	I	1	0	1.12
240213	MAT/EQUIP FOR ROOF/GUTTER OWND	I	1	0	1.12
240214	MAT FOR PANL/ROOF/SIDING,ETC OWNV	I	1	0	1.12
240221	MAT FOR PATIO,MASONRY,ETC RNTR	I	1	0	1.12
240222	MAT FOR PATIO,MASONRY,ETC OWND	I	1	0	1.12
240223	MAT FOR PATIO,MASONRY,ETC OWNV	I	1	0	1.12
240311	PLUMBING SUPP/EQUIP RNTR	I	1	0	1.12
240312	PLUMBING SUPP/EQUIP OWND	I	1	0	1.12
240313	PLUMBING SUPP/EQUIP OWNV	I	1	0	1.12
240321	ELEC SUPP, HEAT/COOL EQUIP RNTR	I	1	0	1.12
240322	ELEC SUPP,HEAT/COOL EQUIP OWND	I	1	0	1.12
240323	ELEC SUPP,HEAT/COOL EQUIP OWNV	I	1	0	1.12
250111	FUEL OIL RNTR	I	1	0	1.39
250112	FUEL OIL OWND	I	1	0	1.39
250113	FUEL OIL OWNV	I	1	0	1.39
250114	FUEL OIL RNTV	I	1	0	1.39
250211	GAS, BOTTLED OR TANK RNTR	I	1	0	1.20
250212	GAS, BOTTLED OR TANK OWND	I	1	0	1.05
250213	GAS,BOTTLED OR TANK - OWNV & RVS	I	1	0	1.05
250214	GAS, BOTTLED OR TANK RNTV	I	1	0	1.05
250221	COAL, RNTR OR OTH PROP	I	1	0	1.12
250222	COAL OWND	I	1	0	1.12
250223	COAL OWNV	I	1	0	1.12
250224	COAL RNTV	I	1	0	1.12
250901	WOOD/KEROSENE/OTHER FUELS RNTR	I	1	0	1.12
250902	WOOD/KEROSENE/OTHER FUELS OWND	I	1	0	1.12
250903	WOOD/KEROSENE/OTHER FUELS OWNV	I	1	0	1.12
250904	WOOD/KEROSENE/OTHER FUELS RNTV	I	1	0	1.12
260111	ELECTRICITY RNTR	I	0	11	0.98
260112	ELECTRICITY OWND	I	0	11	0.85
260113	ELECTRICITY OWNV	I	0	11	0.85
260114	ELECTRICITY RNTV	I	0	11	0.85
260211	UTILITY--NATURAL GAS RNTR	I	0	12	1.20
260212	UTILITY--NATURAL GAS OWND	I	0	12	1.05
260213	UTILITY--NATURAL GAS OWNV	I	0	12	1.05
260214	UTILITY--NATURAL GAS RNTV	I	0	12	1.05
270101	TELEPH SERV, EXCL. MOBILE CAR PH	I	0.5	0	1.12
270102	TELEPH SERVICE FOR MOBILE CAR PH	I	1	0	1.12
270211	WATER AND SEWERAGE MAINT RNTR	I	0	13	1.10
270212	WATER AND SEWERAGE MAINT OWND	I	0	13	1.12
270213	WATER AND SEWERAGE MAINT OWNV	I	0	13	1.12
270214	WATER AND SEWERAGE MAINT RNTV	I	0	13	1.12
270310	COMMUNITY ANTENNA OR CABLE TV	I	0	0	1.12
270411	TRASH/GARBAGE COLLECT RNTR	I	0	14	1.25
270412	TRASH/GARBAGE COLLECT OWND	I	0	14	1.30
270413	TRASH/GARBAGE COLLECT OWNV	I	0	14	1.30
270414	TRASH/GARBAGE COLLECT RNTV	I	0	14	1.30
270901	SEPTIC TANK CLEANING RNTR	I	1	0	1.12

Taxable Items by UCC

270902	SEPTIC TANK CLEANING OWND	I	1	0	1.12
270903	SEPTIC TANK CLEANING OWNV	I	1	0	1.12
270904	SEPTIC TANK CLEANING RNTV	I	1	0	1.12
270905	STEAM HEAT	D	0	0	1.12
280110	BATHROOM LINENS	I	1	0	2.56
280120	BEDROOM LINENS	I	1	0	2.56
280130	KITCHEN AND DINING ROOM LINENS	I	1	0	2.56
280210	CURTAINS AND DRAPES	I	1	0	2.56
280220	SLIPCOVERS/DECORATIVE PILLOWS	I	1	0	2.56
280230	SEWING MATERIALS	I	1	0	2.56
280900	OTHER LINENS	I	1	0	2.56
290110	MATTRESS AND SPRINGS	I	1	0	1.27
290120	OTHER BEDROOM FURNITURE	I	1	0	1.27
290210	SOFAS	I	1	0	1.27
290310	LIVING ROOM CHAIRS	I	1	0	1.27
290320	LIVING ROOM TABLES	I	1	0	1.27
290410	KITCHEN/DINING ROOM FURNITURE	I	1	0	1.27
290420	INFANTS FURNITURE	I	1	0	1.27
290430	OUTDOOR FURNITURE	I	1	0	1.27
290440	WALL UNITS, CABINETS, OCCAS FURN	I	1	0	1.27
300111	PURCH/INST REFRIG/FREEZER RNTR	I	1	0	1.79
300112	PURCH/INST REFRIG/FREEZER OWND	I	1	0	1.79
300211	PURCH/INST CLOTHES WASHER RNTR	I	1	0	1.79
300212	PURCH/INST CLOTHES WASHER OWND	I	1	0	1.79
300221	PURCH/INST CLOTHES DRYER RNTR	I	1	0	1.79
300222	PURCH/INST CLOTHES DRYER OWND	I	1	0	1.79
300311	STOVES, OVENS RNTR	I	1	0	1.79
300312	STOVES, OVENS OWND	I	1	0	1.79
300321	MICROWAVE OVENS RNTR	I	1	0	1.79
300322	MICROWAVE OVENS OWND	I	1	0	1.79
300331	PURCH/INST PORT DISHWASHER RNTR	I	1	0	1.79
300332	PURCH/INST PORT DISHWASHER OWND	I	1	0	1.79
300411	PURCH/INST WINDOW A/C RNTR	I	1	0	1.79
300412	PURCH/INST WINDOW A/C OWND	I	1	0	1.79
310110	BLACK AND WHITE TV	I	1	0	1.61
310120	COLOR TV - CONSOLE	I	1	0	1.61
310130	COLOR TV - PORTABLE/TABLE MOD	I	1	0	1.61
310210	VCRS/VIDEO DISC PLAYERS	I	1	0	1.61
310220	VIDEO CASSETTES/TAPES/DISCS	I	1	0	1.61
310230	VIDEO GAME HARDWARE/SOFTWARE	I	1	0	1.61
310311	RADIOS	I	1	0	1.61
310312	PHONOGRAPHS	I	1	0	1.61
310313	TAPE RECORDERS AND PLAYERS	I	1	0	1.61
310320	COMPONENTS/COMPONENT SYSTEMS	I	1	0	1.61
310333	ACCESSORIES AND OTHER SOUND EQUIP	I	1	0	1.61
310334	SATELLITE DISHES	I	1	0	1.61
310341	RCRD/TAPE/CD/VIDEO MAIL ORD CLUB	I	1	0	1.61
310342	RECORDS,CDS,AUDIO TAPES,NEEDLES	I	1	0	1.61

Taxable Items by UCC

320110	FLOOR COVERINGS (NON-PERM.)	I	1	0	2.56
320111	FLOOR COVERINGS (NON-PERM.)	I	1	0	2.56
320120	WINDOW COVERINGS	I	1	0	2.56
320130	INFANTS EQUIPMENT	I	1	0	1.41
320150	OUTDOOR EQUIPMENT	I	1	0	1.69
320161	NON-INST W/W CARP/SQUARES RNTR	I	1	0	1.69
320162	W/W CARP NON INST REPL/SQS OWND	I	1	0	1.69
320163	W/W CARPET REPL RNTR	I	1	0	1.69
320210	CLOCKS	I	1	0	1.79
320220	LAMPS AND LIGHTING FIXTURES	I	1	0	1.27
320231	OTH HOUSEHOLD DECORATIVE ITEMS	I	1	0	1.27
320232	TELEPHONES AND ACCESSORIES	I	1	0	1.27
320310	PLASTIC DINNERWARE	I	1	0	1.41
320320	CHINA AND OTHER DINNERWARE	I	1	0	1.41
320330	FLATWARE	I	1	0	1.41
320340	GLASSWARE	I	1	0	1.41
320350	SILVER SERVING PIECES	I	1	0	1.41
320360	OTHER SERVING PIECES	I	1	0	1.41
320370	NONELECTRIC COOKWARE	I	1	0	1.41
320380	TABLEWARE/NON-ELEC. KITWARE	D	1	0	1.41
320410	LAWN AND GARDEN EQUIPMENT	I	1	0	1.41
320420	POWER TOOLS	I	1	0	1.41
320430	OTHER HARDWARE	D	1	0	1.41
320511	ELECTRIC FLOOR CLEANING EQUIP	I	1	0	1.41
320512	SEWING MACHINES	I	1	0	1.41
320521	SMALL ELECTRIC KITCHEN APPLIANCES	I	1	0	1.41
320522	PORTABLE HEATING/COOLING EQUIP	I	1	0	1.41
320611	CONSTRCT MAT-RNTR/UDR CN/SEC HM	I	1	0	1.12
320612	CONSTRUCTION MAT OWND	I	1	0	1.12
320613	CONSTRUCTION MAT OWNV	I	1	0	1.12
320621	FLR REP/RPL MAT-RNTR/UDR CN/SC HM	I	1	0	1.69
320622	FLOOR REPAIR/REPL MATERIALS OWND	I	1	0	1.69
320623	FLOOR REPAIR/REPL MATERIALS OWNV	I	1	0	1.69
320631	LNDSCPING MAT-RNTR/UDR CN/SC HM	I	1	0	1.12
320632	LANDSCAPING MATERIALS OWND	I	1	0	1.12
320633	LANDSCAPING MATERIALS OWNV	I	1	0	1.12
320901	OFFICE FURNITURE HOME USE	I	1	0	1.27
320902	HAND TOOLS	I	1	0	1.41
320903	INDOOR PLANTS, FRESH FLOWERS	I	1	0	1.41
320904	CLOSET AND STORAGE ITEMS	I	1	0	1.41
320905	MISC. HOUSEHOLD EQUIP/PARTS	D	1	0	1.41
320906	ELECTRONIC TESTING EQUIP.	D	1	0	1.41
330110	SOAPS AND DETERGENTS	D	1	0	1.12
330210	OTHER LAUNDRY /CLEANING PRODS.	D	1	0	1.12
330310	PAPER TOWELS/NAPKINS/TOILET TI	D	1	0	1.12
330410	STATIONERY, GIFTWRAP, ETC.	D	1	0	1.12
330510	MISC HOUSEHOLD PRODUCTS	D	1	0	1.12
330511	MAT FOR TERMTE/PST CNTRL MAINTCE	I	1	0	1.12

Taxable Items by UCC

330610	LAWN AND GARDEN SUPPLIES	D	1	0	1.12
340110	POSTAGE	D	0	0	1.12
340120	DELIVERY SERVICES	D	1	0	1.12
340211	BABYSIT/CHILD CARE OWN HOME	I	0	0	1.12
340212	BABYSIT/CHILD CARE OTHER HOME	I	0	0	1.12
340310	DOMESTIC SERVICE	I	0	0	1.12
340410	GARDENING/LAWN CARE SERVICE	I	1	0	1.12
340420	WATER SOFTENING SERVICE	I	1	0	1.12
340510	MOVING, STORAGE,FREIGHT EXPRES	I	1	0	1.12
340520	HSHLD LNDRY,DRYCLN NOT COIN-OP	I	1	0	1.12
340530	COIN-OP HSHLD LNDRY, DRY CLN	I	1	0	1.12
340610	REPAIR OF TV/RADIO/SOUND EQUIP	I	1	0	1.61
340620	REPAIR OF HOUSEHOLD APPLIANCES	I	1	0	1.41
340630	REUPHOLSTERY OF FURNITURE	I	1	0	1.27
340901	RENTAL/REPAIR-TOOLS,LAWN/GARDEN	I	1	0	1.41
340902	RENTAL OF TELEVISIONS	I	1	0	1.61
340903	MISC. HOME SERVICES	I	1	0	1.12
340904	RENTAL OF FURNITURE	I	1	0	1.39
340905	RENTAL OF VCR/RADIO/SOUND EQUIP	I	1	0	1.61
340906	CARE OF INVALIDS, ELDERLY, ETC	I	0	0	1.12
340907	RENTAL OF HOUSEHOLD EQUIPMENT	I	1	0	1.41
340908	RNTL OFF EQUIP NON-BUS USE	I	1	0	1.41
340910	ADULT DAY CARE CENTERS	I	0	0	1.12
340911	MNGMT/SPEC SER/SECURITY OWND	I	0	0	1.12
340912	MNGMT/SPEC SER/SECURITY OWNV	I	0	0	1.12
340913	REPAIR OF MISC HSHLD EQ/FSHGS	D	1	0	1.12
340914	SERV FOR TERMT/PST CNTRL	I	1	0	1.12
340915	HOME SECURITY SYS. SERV. FEE	I	1	0	1.12
350110	TENANTS INSURANCE	I	0	7	1.12
360110	MENS SUITS	I	1	0	1.25
360120	MENS SPORTCOATS/TAILORED JACKETS	I	1	0	1.25
360210	MENS COATS AND JACKETS	I	1	0	1.25
360311	MENS UNDERWEAR	I	1	0	1.25
360312	MENS HOSIERY	I	1	0	1.25
360320	MENS NIGHTWEAR/LOUNGEWEAR	I	1	0	1.25
360330	MENS ACCESSORIES	I	1	0	1.25
360340	MENS SWEATERS AND VESTS	I	1	0	1.25
360350	MENS ACTIVE SPORTSWEAR	I	1	0	1.25
360410	MENS SHIRTS	I	1	0	1.25
360511	MENS PANTS	I	1	0	1.25
360512	MENS SHORTS/SHORTS SETS	I	1	0	1.25
360901	MENS UNIFORMS	I	1	0	1.25
360902	MENS COSTUMES	I	1	0	1.25
370110	BOYS COATS AND JACKETS	I	1	0	1.25
370120	BOYS SWEATERS	I	1	0	1.25
370130	BOYS SHIRTS	I	1	0	1.25
370211	BOYS UNDERWEAR	I	1	0	1.25
370212	BOYS NIGHTWEAR	I	1	0	1.25

Taxable Items by UCC

370213	BOYS HOSIERY	I	1	0	1.25
370220	BOYS ACCESSORIES	I	1	0	1.25
370311	BOYS SUITS, SPORTCOATS,VESTS	I	1	0	1.25
370312	BOYS PANTS	I	1	0	1.25
370313	BOYS SHORTS, SHORTS SETS	I	1	0	1.25
370902	BOYS COSTUMES	I	1	0	1.25
370903	BOYS UNIFORMS	I	1	0	1.25
370904	BOYS ACTIVE SPORTSWEAR	I	1	0	1.25
380110	WOMENS COATS AND JACKETS	I	1	0	1.25
380210	WOMENS DRESSES	I	1	0	1.25
380311	WOMENS SPORTCOATS, TAIL. JKTS	I	1	0	1.25
380312	WOMENS VESTS AND SWEATERS	I	1	0	1.25
380313	WOMENS SHIRTS, TOPS,BLOUSES	I	1	0	1.25
380320	WOMENS SKIRTS	I	1	0	1.25
380331	WOMENS PANTS	I	1	0	1.25
380332	WOMENS SHORTS,SHORTS SETS	I	1	0	1.25
380340	WOMENS ACTIVE SPORTSWEAR	I	1	0	1.25
380410	WOMENS SLEEPWEAR	I	1	0	1.25
380420	WOMENS UNDERGARMENTS	I	1	0	1.25
380430	WOMENS HOSIERY	I	1	0	1.25
380510	WOMENS SUITS	I	1	0	1.25
380901	WOMENS ACCESSORIES	I	1	0	1.25
380902	WOMENS UNIFORMS	I	1	0	1.25
380903	WOMENS COSTUMES	I	1	0	1.25
390110	GIRLS COATS AND JACKETS	I	1	0	1.25
390120	GIRLS DRESSES, SUITS	I	1	0	1.25
390210	GIRLS SHIRTS/BLOUSES/SWEATERS	I	1	0	1.25
390221	GIRLS SKIRTS AND PANTS	I	1	0	1.25
390222	GIRLS SHORTS, SHORTS SETS	I	1	0	1.25
390230	GIRLS ACTIVE SPORTSWEAR	I	1	0	1.25
390310	GIRLS UNDERWEAR AND SLEEPWEAR	I	1	0	1.25
390321	GIRLS HOSIERY	I	1	0	1.25
390322	GIRLS ACCESSORIES	I	1	0	1.25
390901	GIRLS UNIFORMS	I	1	0	1.25
390902	GIRLS COSTUMES	I	1	0	1.25
400110	MENS FOOTWEAR	I	1	0	1.25
400210	BOYS FOOTWEAR	I	1	0	1.25
400220	GIRLS FOOTWEAR	I	1	0	1.25
400310	WOMENS FOOTWEAR	I	1	0	1.25
410110	INFANT COAT/JACKET/SNOWSUIT	I	1	0	1.25
410120	INFANT DRESSES/OUTERWEAR	I	1	0	1.25
410130	INFANT UNDERGARMENTS	I	1	0	1.25
410140	INFANT NIGHTWEAR/LOUNGEWEAR	I	1	0	1.25
410901	INFANTS ACCESSORIES	I	1	0	1.25
420110	MATERIAL FOR MAKING CLOTHES	I	1	0	1.25
420120	SEWING NOTIONS, PATTERNS	I	1	0	1.25
430110	WATCHES	I	1	0	1.25
430120	JEWELRY	I	1	0	1.25

Taxable Items by UCC

430130	LUGGAGE	I	1	0	1.25
440110	SHOE REPAIR, OTH SHOE SERVICE	I	1	0	1.25
440120	COIN-OP APPAREL LDRY/DRY CLNG	I	1	0	1.25
440130	ALTER/REPAIR OF APPAREL, ACCESS	I	1	0	1.25
440140	CLOTHING RENTAL	I	1	0	1.25
440150	WATCH AND JEWELRY REPAIR	I	1	0	1.25
440210	APPAREL LNDRY/DRY CLNG N/COIN-OP	I	1	0	1.25
440900	CLOTHING STORAGE	I	1	0	1.25
450110	NEW CARS	I	1	0	0.97
450116	TRADE-IN ALLOWANCE/NEW CARS	I	0	0	0.97
450210	NEW TRUCKS	I	1	0	0.97
450216	TRADE-IN ALLOW/NEW TRUCKS	I	0	0	0.97
450220	NEW MOTORCYCLES	I	1	0	0.97
450226	TRADE-IN ALLOW/NEW MOTORCYCLES	I	0	0	0.97
450310	CAR LEASE PAYMENTS	I	1	0	0.97
450311	EXTRA FEES,INSUR,MAINT/CAR LEASE	I	1	0	0.97
450312	TRADE-IN ALLOWANCE/CAR LEASE	I	0	0	0.97
450313	CASH DOWNPAYMENT/ CAR LEASE	I	0	0	0.97
450314	TERMINATION FEE/CAR LEASE	I	1	0	0.97
450410	TRUCK LEASE PAYMENTS	I	1	0	0.97
450411	EXTRA FEES,INS,MAINT/TRUCK LEASE	I	1	0	0.97
450412	TRADE-IN ALLOWANCE/TRUCK LEASE	I	0	0	0.97
450413	CASH DOWNPAYMENT/TRUCK LEASE	I	0	0	0.97
450414	TERMINATION FEE/TRUCK LEASE	I	1	0	0.97
450900	AIRCRAFT	D	1	0	0.97
460110	USED CARS	I	1	0	0.97
460116	TRADE-IN ALLOWANCE/USED CARS	I	0	0	0.97
460901	USED TRUCKS	I	1	0	0.97
460902	USED MOTORCYCLES	I	1	0	0.97
460907	TRADE-IN ALLOWANCE/USED TRUCKS	I	0	0	0.97
460908	TRADE-IN ALLOW/USED MOTORCYCLES	I	0	0	0.97
470111	GASOLINE	I	0	15	1.18
470112	DIESEL FUEL	I	0	15	1.18
470113	GASOLINE ON OUT OF TOWN TRIPS	I	0	15	1.18
470114	GASAHOL	D	0	0	1.18
470211	MOTOROIL	I	1	0	1.04
470212	MOTOR OIL ON OUT OF TOWN TRIPS	I	1	0	1.04
470220	COOLANT/ADDITIVES/BRK/TRNS FLD	I	1	0	1.72
480110	TIRES PURCHASED/REPLACED/INSTALL	I	1	0	1.72
480212	VEHICLE PRODUCTS	D	1	0	1.72
480213	PARTS/EQUIP/ACCESSORIES	I	1	0	1.72
480214	VEHICLE AUDIO EQ. EXCL. LABOR	I	1	0	1.72
490000	MISC. AUTO REPAIR/SERVICING	D	1	0	1.72
490110	BODY WORK AND PAINTING	I	1	0	1.72
490211	CLUTCH, TRANSMISSION REPAIR	I	1	0	1.72
490212	DRIVE SHAFT AND REAR-END REPAIR	I	1	0	1.72
490221	BRAKE WORK	I	1	0	1.72
490231	REPAIR TO STEERING OR FRONT END	I	1	0	1.72

Taxable Items by UCC

490232	REPAIR TO ENGINE COOLING SYSTEM	I	1	0	1.72
490311	MOTOR TUNE-UP	I	1	0	1.72
490312	LUBE, OIL CHANGE AND OIL FILTERS	I	1	0	1.72
490313	FRNT END ALIGN, WHEEL BAL/ROTAT	I	1	0	1.72
490314	SHOCK ABSORBER REPLACEMENT	I	1	0	1.72
490316	GAS TANK REPAIR,REPLACEMENT	D	1	0	1.72
490318	TIRE REPAIR AND OTH REPAIR WORK	I	1	0	1.72
490319	VEHICLE AIR CONDITION REPAIR	I	1	0	1.72
490411	EXHAUST SYSTEM REPAIR	I	1	0	1.72
490412	ELECTRICAL SYSTEM REPAIR	I	1	0	1.72
490413	MOTOR REPAIR/REPLACEMENT	I	1	0	1.72
490501	VEHICLE ACCESSORIES INCL. LABOR	I	1	0	1.72
490502	VEHICLE AUDIO EQ. INCL. LABOR	I	1	0	1.72
490900	AUTO REPAIR SERVICE POLICY	I	1	0	1.72
500110	VEHICLE INSURANCE	I	0	7	1.72
510110	AUTOMOBILE FINANCE CHARGES	I	0	0	1.72
510901	TRUCK FINANCE CHARGES	I	0	0	1.72
510902	MOTORCYCLE & PLANE FINANCE CHG	I	0	0	1.72
520110	STATE/LOCAL REGISTRATION	I	0	0	1.72
520310	DRIVERS LICENSE	I	0	0	1.72
520410	VEHICLE INSPECTION	I	0	0	1.72
520511	AUTO RENTAL	I	1	16	1.72
520512	AUTO RENTAL, OUT-OF-TOWN TRIPS	I	0.705	16	1.72
520521	TRUCK RENTAL	I	1	0	1.72
520522	TRUCK RENTAL, OUT-OF-TOWN TRIP	I	0.705	0	1.72
520531	PRKNG FEE IN HME CITY EXCL RSDNC	I	1	0	1.72
520532	PARKING FEES, OUT-OF-TOWN TRIP	I	0.9	0	1.72
520541	TOLLS	D	0	0	1.72
520542	TOLLS ON OUT-OF-TOWN TRIPS	I	0	0	1.72
520550	TOWING CHARGES	I	1	0	1.72
520901	DOCKING/LANDING FEES	I	0	0	1.72
520902	MOTORCYCLE RENTAL	I	1	0	1.72
520903	AIRCRAFT RENTAL	I	1	0	1.72
520904	RENTAL NON-CAMPER TRAILER	I	1	0	1.72
520905	MOTORCYCLE RENTAL OUT-OF-TOWN	I	0.705	0	1.72
520906	AIRCRAFT RENTAL/OUT-OF-TOWN TR	I	0.705	0	1.72
520907	BOAT/TRAILER RENT OUT OF TOWN	I	0.705	0	1.72
530110	AIRLINE FARES	I	0	0	1.64
530210	INTERCITY BUS FARES	I	0	17	1.64
530311	INTRACITY MASS TRANSIT FARES	I	0	18	1.64
530312	LOCAL TRANS. OUT OF TOWN TRIPS	I	0	17	1.64
530411	TAXI FARES ON TRIPS	I	0	18	1.64
530412	TAXI FARES AND LIMOUSINE SERVICE	I	0	18	1.64
530510	INTERCITY TRAIN FARES	I	0	17	1.64
530901	SHIP FARES	I	0	0	1.64
530902	SCHOOL BUS	I	0	0	1.64
530903	CAR/VAN POOL & NON-MOTOR TRANS	D	0	0	1.64
540000	PRESCRIPTION DRUGS	I	0	0	1.64

Taxable Items by UCC

550110	EYEGLASSES AND CONTACT LENSES	I	0	0	1.64
550210	OVER-THE-COUNTER DRUGS	D	1	0	1.64
550310	TOPICALS AND DRESSINGS	D	1	0	1.64
550320	MEDICAL EQUIP. FOR GENERAL USE	I	1	0	1.64
550330	SUPPORTIVE/CONVAL MED. EQUIP.	I	1	0	1.64
550340	HEARING AIDS	I	0	0	1.64
550410	NONPRESCRIP VITAMINS	D	1	0	1.64
550900	RECREATIONAL DRUGS	D	0	0	1.85
560110	PHYSICIANS SERVICES	I	0	0	1.49
560210	DENTAL SERVICES	I	0	0	1.49
560310	EYECARE SERVICES	I	0	0	1.49
560330	LAB TESTS, X-RAYS	I	0	0	1.49
560400	SERV BY PROS OTH THAN PHYSICIANS	I	0	0	1.49
570110	HOSPITAL ROOM	I	0	0	1.15
570210	HOSPITAL SERVICE OTH THAN ROOM	I	0	0	1.15
570220	CARE IN CONVL OR NURSING HOME	I	0	0	1.15
570230	OTHER MEDICAL CARE SERVICE	I	0	0	1.15
570240	MEDCL SERV INCL IN HM OWN EXP,3I	I	0	0	1.15
570901	RENTAL OF MEDICAL/SURGICAL EQUIP	I	1	0	1.00
570902	REPAIR OF MEDICAL EQUIPMENT	D	1	0	1.00
570903	RENTAL OF SUPORTIVE/CONVAL EQUIP	I	1	0	1.00
580111	TRD FEE FOR SRV HLTH P (NO BCBS)	I	0	7	1.00
580112	TRD FEE FOR SRV HLTH P (BCBS)	I	0	7	1.00
580113	PREF PROVIDER HLTH PLN (NO BCBS)	I	0	7	1.00
580114	PREF PROVIDER HLTH PLN (BCBS)	I	0	7	1.00
580311	HLTH MAINT. ORG (NO BCBS)	I	0	7	1.00
580312	HLTH MAINT. ORG (BCBS)	I	0	7	1.00
580901	MEDICARE PAYMENTS	I	0	0	1.00
580903	COML MEDICARE SUPPLEMNT (NO BCBS)	I	0	7	1.00
580904	COML MEDICARE SUPPLEMENT (BCBS)	I	0	7	1.00
580905	OTHER HEALTH INSURANCE (NO BCBS)	I	0	7	1.00
580906	OTHER HEALTH INSURANCE (BCBS)	I	0	7	1.00
590111	NEWSPAPER SUBSCRIPTIONS	I	0	0	2.22
590112	NEWSPAPERS, NON-SUBSCRIPTION	I	0	0	2.22
590211	MAGAZINE SUBSCRIPTIONS	I	1	0	2.22
590212	MAGAZINES, NON-SUBSCRIPTION	I	1	0	2.22
590220	BOOKS THRU BOOK CLUBS	I	1	0	2.22
590230	BOOKS NOT THRU BOOK CLUBS	I	1	0	2.22
590900	NEWSLETTERS	D	1	0	2.22
600110	OUTBOARD MOTOR	I	1	0	2.27
600121	BOAT W/O MOTOR/BOAT TRAILERS	I	1	0	2.27
600122	TRAIL/OTH ATTACHABLE CAMPERS	I	1	0	2.27
600127	TRADE ALLOW/BOATS W/O MOTOR/TRAIL	I	0	0	2.27
600128	TRADE ALLOW/TRAILERS, ATT CAMPERS	I	0	0	2.27
600132	PURCHASE OF BOAT WITH MOTOR	I	1	0	2.27
600138	TRADE-IN ALLOW/BOATS WITH MOTORS	I	0	0	2.27
600141	PURCHASE OF MOTORIZED CAMPER	I	1	0	2.27
600142	PURCHASE OF OTHER VEHICLE	I	1	0	2.27

Taxable Items by UCC

600143	TRADE ALLOW MOTORIZED CAMPER	I	0	0	2.27
600144	TRADE ALLOW OTHER VEHICLE	I	0	0	2.27
600210	GENERAL SPORT/EXCERCISE EQUIP	I	1	0	2.27
600310	BICYCLES	I	1	0	2.27
600410	CAMPING EQUIPMENT	I	1	0	2.27
600420	HUNTING, FISHING EQUIPMENT	I	1	0	2.27
600430	WINTER SPORT EQUIPMENT	I	1	0	2.27
600901	WATER SPORT EQUIPMENT	I	1	0	2.27
600902	OTHER SPORT EQUIPMENT	I	1	0	2.27
610110	TOYS GAMES HOBBIES TRICYCLES	I	1	0	2.27
610120	PLAYGROUND EQUIPMENT	I	1	0	2.27
610130	MUSIC INSTRUMENTS/ACCESSORIES	I	1	0	2.27
610210	FILM	I	1	0	2.27
610220	OTHER PHOTOGRAPHIC SUPPLIES	D	1	0	2.27
610230	PHOTOGRAPHIC EQUIPMENT	I	1	0	2.27
610310	PET FOOD	D	1	0	1.47
610320	PET-PURCHASE/SUPPLIES/MEDICINE	I	1	0	1.47
610900	REC EXPNS OUTSIDE HOME CITY	I	0.8	0	1.47
610901	FIREWORKS	D	1	0	1.64
610902	SOUVENIRS	D	1	0	1.64
610903	VISUAL GOODS	D	1	0	1.64
620111	SOCIAL/RECRE/CIVIC CLUB MEMBRSHP	I	0	0	1.64
620112	CREDIT CARD MEMBERSHIPS	I	0	0	1.64
620113	AUTOMOBILE SERVICE CLUBS	I	0	0	1.64
620121	FEES FOR PARTICIPANT SPORTS	I	1	0	1.39
620122	PARTIC. SPORTS OUT-OF-TOWN TRI	I	0.8	0	1.39
620211	MOVIE, THEATER, OPERA, BALLET	I	0	0	1.39
620212	MOVIE, OTH ADM. OUT-OF-TOWN	I	0	0	1.39
620221	ADMISSION TO SPORTING EVENTS	I	0	0	1.39
620222	ADM TO SPRTS EVENTS OUT-OF-TOW	I	0	0	1.39
620310	FEES FOR RECREATIONAL LESSONS	I	1	0	1.39
620320	PHOTOGRAPHER FEES	I	1	0	2.27
620330	FILM PROCESSING	I	1	0	2.27
620410	PET SERVICES	I	1	0	1.47
620420	VET SERVICES	I	1	0	1.47
620510	ADMISSIONS MISC	D	0	0	1.39
620610	MISC. ENTERTAINMENT SERVICES	D	1	0	1.39
620710	CAMP FEES	D	0.1	0	1.39
620903	OTH ENT SERV, OUT-OF-TOWN TRIP	I	0.8	0	1.39
620904	RENT/REP MUSIC INSTRUMENTS	I	1	0	1.64
620905	RENT/REP PHOTO EQUIP	I	1	0	1.64
620906	RENTAL OF BOAT	I	1	0	1.64
620908	RENT/REPAIR OF MISC SPORTS EQU	I	1	0	1.64
620909	RENTAL CAMPER ON TRIPS	I	0.705	0	1.64
620911	MISC FEES,PARIMUTEL LOSSES	D	0	0	1.64
620912	RNTL VIDEO CASS/TAPES/DISCS/FILMS	I	1	0	1.64
620913	PINBALL/ELECTRONIC VIDEO GAMES	D	1	0	1.64
620919	RENTAL OTHER VEHICLES ON TRIPS	I	0.705	0	1.64

Taxable Items by UCC

620921	RENTAL OF MOTORIZED CAMPER	I	0.705	0	1.64
620922	RENTAL OF OTHER RVS	I	0.705	0	1.64
630110	CIGARETTES	I	1	8	1.43
630210	OTHER TOBACCO PRODUCTS	I	1	9	1.28
630220	SMOKING ACCESSORIES	D	1	0	1.43
630900	MARIJUANA	D	0	0	1.85
640110	HAIR CARE PRODUCTS	D	1	0	1.64
640120	NON-ELEC ARTICLES FOR THE HAIR	D	1	0	1.64
640130	WIGS AND HAIRPIECES	I	1	0	1.64
640220	SHAVING NEEDS	D	1	0	1.64
640310	COSMETICS, PERFUME, BATH PREP	D	1	0	1.64
640410	DEOD,FEM HYG, MISC. PERS. CARE	D	1	0	1.64
640420	ELECTRIC PERSONAL CARE APPL.	I	1	0	1.64
650110	PERS. CARE SERV FOR FEMALES	I	1	0	1.64
650210	PERS. CARE SERV FOR MALES	I	1	0	1.64
650310	PERS. CARE SERV.	I	1	0	1.64
650900	REPAIR OF PERS. CARE APP.	I	1	0	1.64
660000	SCHOOL SUPPL., ETC. - UNSPEC.	D	1	0	2.22
660110	SCHOOL BK/SUPL/EQUIP FOR COLLEGE	I	1	0	2.22
660210	SCHOOL BK/SUPL/EQUIP FOR ELEM/HS	I	1	0	2.22
660310	ENCYL. OTH SETS OF REFRNCE BKS	I	1	0	2.22
660900	SCH BK/SUP/EQ-DAY CARE,NURS,OTH	I	1	0	2.22
670110	COLLEGE TUITION	I	0	0	1.00
670210	ELEM./H.S. TUITION	I	0	0	1.00
670310	DAY CARE/NURS/PRSCH EXP INCL TUIT	I	0	0	1.00
670901	OTHER SCHOOL TUITION	I	0	0	1.00
670902	OTH SCH EXPENSES INCLUD RENTALS	I	1	0	1.00
680110	LEGAL FEES	I	0	0	1.00
680140	FUNERAL EXPENSE	I	0.5	0	1.00
680210	SAFE DEPOSIT BOX RENTAL	I	0	0	1.00
680220	CHECK ACCTS / OTH BANK SERV CHGS	I	0	0	3.33
680901	CEMETERY LOTS,VAULTS,MAINT FEES	I	0	0	3.33
680902	ACCOUNTING FEES	I	0	0	3.33
680903	MISC. PERS. SERVICES	D	0	0	3.33
690111	COMPTER/COMPTER HRDWAR N-BUS USE	I	1	0	1.41
690112	COMPTER SFTWR/CMPTR ACC N-BUS USE	I	1	0	1.41
690113	REPAIR-CMPTR,CMPTR SYS N-B	I	1	0	1.41
690114	COMPUTER INFORMATION SERVICES	I	1	0	1.41
690210	TELEPHONE ANSWERING DEVICES	I	1	0	1.41
690220	CALCULATORS	I	1	0	1.41
690230	TYPWRITS/OTH OFF MACH NON-BUS USE	I	1	0	1.41
690241	SMOKE ALARM PUR/RENT RNTR	I	1	0	1.41
690242	SMOKE ALARM PUR/RENT OWND	I	1	0	1.41
690243	SMOKE ALARM PUR/RENT OWNV	I	1	0	1.41
690244	OTH HH APPL RNTR	I	1	0	1.41
690245	OTH HH APPL OWND	I	1	0	1.41
700110	LIFE/ENDOW/ANNUIT/OTH PERS INS	I	0	7	1.00
710110	FINANCE CHARGES EXCL MORT/VEH	I	0	0	1.00

Taxable Items by UCC

790210	FOOD/GOODS/BEV AT GROC STORES	I	0	0	1.35
790220	FOOD/NONALC BEV AT GROC STORES	I	0	0	1.35
790230	FD/NONALC BEV AT CONVEN STORE	I	0	0	1.35
790310	BEER/WINE FOR HOME USE	I	0	0	2.56
790320	OTHER ALCOHOL FOR HOME USE	I	0	0	2.56
790410	DINING OUT AT REST., ETC EXCL ALC	I	0	0	2.56
790420	ALCOHOL AT RESTAURANTS ETC	I	0	0	2.56
790430	SCHOOL MEALS	I	0	0	1.30
790600	MAINT/REP/UTIL OTH PROP	I	1	0	1.12
790610	CAPITAL IMPROVEMENTS OTH PROP	I	1	0	1.12
790611	DWASH/DISP/HOOD CAP IMPR OTH PROP	I	1	0	1.12
790620	MGMT SERVICES/IMP OF OTH PROP	I	0	0	1.12
790630	SPECIAL ASSESSMENTS OTH PROP	I	0	0	1.12
790640	MGMT/SECURITY/PARKING OTH PROP	I	0	0	1.12
790690	CONSTR MAT JOB NOT STRTD RNTR	I	1	0	1.12
790710	PURCHASE OTHER PROPERTY	I	0	0	1.00
790730	ORIG MORT AMT OTH PROP	I	0	0	1.00
790810	SALE PRICE OF OTHER PROP	I	0	10	1.00
790820	MORT HELD AFTER SALE OTH PROP	I	0	0	1.00
790830	TOT EXP IN SALE OF OTH PROP	I	0	0	1.00
790910	SPECIAL/LUMP MORT PAY OTH PROP	I	0	10	1.00
790920	REDUCT MORT PRIN OTH PROP	I	0	0	1.00
790930	ORIG MORT AMT OTH PROP	I	0	0	1.00
790940	RDCT PRIN,LMP SM HM EQ LN,OTH PR	I	0	0	1.00
790950	ORIG LN AMT,LP SM HM EQ LN,OT PR	I	0	0	1.00
800111	ALIMONY QUARTERLY S19	I	0	0	1.00
800112	ALIMONY ANNUAL S22	I	0	0	1.00
800121	CHILD SUPPORT QUARTERLY S19	I	0	0	1.00
800122	CHILD SUPPORT ANNUAL S22	I	0	0	1.00
800700	MEALS AS PAY	I	0	0	1.00
800710	RENT AS PAY	I	0	0	1.00
800721	MARKET VALUE OF OWNED HOME	I	0	0	1.00
800801	CASH SUP N/CU INCL STUD ALIM CHLD	I	0	0	1.00
800802	CASH SUPPORT COLL STUDENT	I	0	0	1.00
800803	CSH GFT/NON-CU, CNTRB/ORG	I	0	0	1.00
800810	GIFTS N/CU - CASH, BONDS, STOCKS	I	0	0	1.00
800820	CONTRIBUTIONS TO CHARITY	I	0	0	1.00
800830	CONTRIBUTIONS TO CHURCH	I	0	0	1.00
800840	CONTRIBUTIONS TO EDUCATION INST	I	0	0	1.00
800850	CONTRIBUTIONS TO POLITICAL ORGS	I	0	0	1.00
800860	OTHER CONTRIBUTIONS	I	0	0	1.00
800910	DEDUCTIONS FOR GOVT RETIREMENT	I	0	0	1.00
800920	DEDUCTIONS FOR RR RETIREMENT	I	0	0	1.00
800931	DEDUCTIONS FOR PRIVATE PENSIONS	I	0	0	1.00
800932	NON-PAYROL DEP TO RETIRE PLAN	I	0	0	1.00
800940	DEDUCTIONS FOR SOCIAL SECURITY	I	0	0	1.00
810101	PURCHASE PROPERTY, OWND	I	0	0	1.00
810102	PURCHASE PROPERTY, OWNV	I	0	0	1.00

Taxable Items by UCC

810301	CLOSING COSTS OWND	I	0	0	1.00
810302	CLOSING COSTS OWNV	I	0	0	1.00
810400	GIFT OF TRIP EXPENSES	I	0	0	1.00
820101	SELLING PRICE OWND	I	0	0	1.00
820102	SELLING PRICE OWNV	I	0	0	1.00
820201	PRIN AMT TRUST HELD OWND	I	0	0	1.00
820202	PRIN AMT TRUST HELD OWNV	I	0	0	1.00
820301	TOTAL SELLING EXP OWND	I	0	10	1.00
820302	TOTAL SELLING EXP OWNV	I	0	10	1.00
830101	SPEC/LUMP MORT PAY OWND	I	0	0	1.00
830102	SPEC/LUMP MORT PAY OWNV	I	0	0	1.00
830201	REDUCTION MORTGAGE PRINC OWND	I	0	0	1.00
830202	REDUCTION MORTGAGE PRINC OWNV	I	0	0	1.00
830203	RDUCT PRINC,LMP SM HM EQ LN,OWND	I	0	0	1.00
830204	RDUCT PRINC,LMP SM HM EQ LN,OWNV	I	0	0	1.00
830301	ORIGINAL MORT AMT OWND	I	0	0	1.00
830302	ORIGINAL MORT AMT OWNV	I	0	0	1.00
830303	ORIG LN AMT,LMP SM HM EQ LN,OWND	I	0	0	1.00
830304	ORIG LN AMT,LMP SM HM EQ LN,OWNV	I	0	0	1.00
840101	AMT FOR SPECIAL ASSESMENT OWND	I	0	0	1.00
840102	AMT FOR SPECIAL ASSESMENT OWNV	I	0	0	1.00
850100	REDUCTION PRINC VEH LOAN	I	0	0	1.00
850200	AMT BORROWED EXLC INT VEH LOAN	I	0	0	1.00
850300	INT PAID ON OTH VEH	I	0	0	1.00
860100	AMT AUTO SOLD OR REIMBURSED	I	0	0	1.00
860200	AMT TRUCK SOLD OR REIMBURSED	I	0	0	1.00
860301	AMT MOTOR CAMPER SOLD/REIMB	I	0	0	1.00
860302	AMT OTH VEHICLE SOLD/REIMB	I	0	0	1.00
860400	AMT NONMOTOR CAMPER SOLD/REIM	I	0	0	1.00
860500	AMT MOTORCYCLE SOLD/REIM	I	0	0	1.00
860600	AMT BOAT WITH MOTOR SOLD/REIM	I	0	0	1.00
860700	AMT BOAT W/O MOTOR SOLD/REIM	I	0	0	1.00
860800	AMT PLANE SOLD/REIM	I	0	0	1.00
870101	NEW CARS/TRUCKS, NOT FIN.	I	0	0	1.00
870102	DOWNPAY, NEW CARS/TRUCKS, FIN.	I	0	0	1.00
870103	INTEREST, NEW CARS/TRUCKS, FIN.	I	0	0	1.00
870104	PRINCIPAL, NEW CARS/TRUCKS, FIN.	I	0	0	1.00
870201	USED CARS/TRUCKS, NOT FIN.	I	0	0	1.00
870202	DOWNPAY, USED CARS/TRUCKS, FIN.	I	0	0	1.00
870203	INTEREST, USED CARS/TRUCKS, FIN.	I	0	0	1.00
870204	PRINCIPAL, USED CARS/TRUCKS, FIN.	I	0	0	1.00
870301	MOTORCYCLES/AIRCRAFT, NOT FIN.	I	0	0	1.00
870302	DOWNPAY, CYCLES/PLANES, FIN.	I	0	0	1.00
870303	INTEREST, CYCLES/PLANES, FIN.	I	0	0	1.00
870304	PRINCIPAL, CYCLES/PLANES, FIN.	I	0	0	1.00
870401	BOAT/TRAILERS, NOT FIN.	I	0	0	1.00
870402	DOWNPAY, BOAT/TRAILERS, FIN.	I	0	0	1.00
870403	INTEREST, BOAT/TRAILERS, FIN.	I	0	0	1.00

Taxable Items by UCC

870404	PRINCIPAL, BOAT/TRAILERS, FIN.	I	0	0	1.00
870501	UNMOTOR CAMPER, NOT FIN.	I	0	0	1.00
870502	DOWNPAY, UNMOTOR CAMPER, FIN.	I	0	0	1.00
870503	INTEREST, UNMOTOR CAMPER, FIN.	I	0	0	1.00
870504	PRINCIPAL, UNMOTOR CAMPER, FIN.	I	0	0	1.00
870605	PURCHASE OF MTR CAMPER, NOT FIN.	I	0	0	1.00
870606	PRINCIPAL, MTR CAMPER, FIN.	I	0	0	1.00
870607	INTEREST, MOTORIZED CAMPER, FIN.	I	0	0	1.00
870608	DOWNPAY, MTR CAMPER, FIN.	I	0	0	1.00
870701	MOTORBOATS, NOT FIN.	I	0	0	1.00
870702	DOWNPAY, MOTORBOATS, FIN.	I	0	0	1.00
870703	INTEREST, MOTORBOATS, FIN.	I	0	0	1.00
870704	PRINCIPAL, MOTORBOATS, FIN.	I	0	0	1.00
870801	PURCH OF OTHER VEHICLE, NOT FIN	I	0	0	1.00
870802	PRINCIPAL, OTHER VEHICLE, FIN.	I	0	0	1.00
870803	INTEREST, OTHER VEHICLE, FIN.	I	0	0	1.00
870804	DOWNPAY, OTH VEHICLE, FIN.	I	0	0	1.00
880100	TOT PAID, HM EQ LN (CRDT), OWND	I	0	0	1.00
880110	INTEREST, HM EQ LN (CRDT), OWND	I	0	0	1.00
880120	PRNCPL, HM EQ LN (CRDT), OWND	I	0	0	1.00
880200	TOT PD, HM EQ LN(CRDT), OTH PROP	I	0	0	1.00
880210	INTERST,HM EQ LN(CRDT), OTH PROP	I	0	0	1.00
880220	PRNCPL, HM EQ LN (CRDT), OTH PR	I	0	0	1.00
880300	TOT PAID, HM EQ LN (CRDT), OWNV	I	0	0	1.00
880310	INTEREST, HM EQ LN (CRDT), OWNV	I	0	0	1.00
880320	PRNCPL, HM EQ LOAN (CRDT), OWNV	I	0	0	1.00
900000	WAGES AND SALARIES	I	0	0	1.00
900001	OCCUPATIONAL EXPENSES	I	0	0	1.00
900010	NET BUSINESS INCOME	I	0	0	1.00
900020	NET FARM INCOME	I	0	0	1.00
900030	SS/RRR INCOME	I	0	0	1.00
900040	PENSIONS AND ANNUITIES	I	0	0	1.00
900050	DIVIDEND/ROYALTY/ESTATE/TRUST	I	0	0	1.00
900060	ROOMER AND BOARDER INCOME	I	0	0	1.00
900070	OTHER RENTAL INCOME	I	0	0	1.00
900080	INTEREST	I	0	0	1.00
900090	SUPPLEMENTAL SECURITY INCOME	I	0	0	1.00
900100	UNEMPLOYMENT COMPENSATION	I	0	0	1.00
900110	WORKERS COMPENSATION	I	0	0	1.00
900120	PUBLIC ASSISTANCE	I	0	0	1.00
900130	REGULAR CONTRIBUTIONS FOR SUPPORT	I	0	0	1.00
900131	OTHER CHILD SUPPORT PAYMENT	I	0	0	1.00
900132	REG CONTRIBUTION OTHER SOURCES	I	0	0	1.00
900140	OTHER INCOME	I	0	0	1.00
900150	FOOD STAMPS	I	0	0	1.00
910000	LUMP SUM PAYMENTS	I	0	0	1.00
910010	MONEY FROM SALE HH FURNS ETC	I	0	0	1.00
910020	OVERPAYMENT ON SOCIAL SECURITY	I	0	0	1.00

Taxable Items by UCC

910030	REFUND FROM INSURANCE POLICIES	I	0	0	1.00
910040	REFUNDS FROM PROPERTY TAXES	I	0	0	1.00
910041	LUMP SUM CHILD SUPPORT PAYMENT	I	0	0	1.00
910050	RENTAL EQUIVALENCE OF OWNED HOME	I	0	0	1.00
910060	EST RNT VALU TIME SHARE NOT RNTD	I	0	0	1.00
910070	EST RNT VALU NONTIM SHR NOT RNT	I	0	0	1.00
910080	RENT RECEIVED TIME SHARE	I	0	0	1.00
910090	RENT RECEIVED NONTIME SHARE	I	0	0	1.00
910100	RENTAL EQUIVALENCE OF VAC HOME	I	0	0	1.00
920010	MARKET VAL SAVINGS ACCOUNTS	I	0	0	1.00
920020	MARKET VAL CHECKING ACCOUNTS	I	0	0	1.00
920030	MARKET VAL US SAVINGS BONDS	I	0	0	1.00
920040	MARKET VAL ALL SECURITIES	I	0	0	1.00
950000	FEDERAL INCOME TAX	I	0	0	1.00
950001	FEDERAL INCOME TAX REFUNDS	I	0	0	1.00
950010	STATE/LOCAL INCOME TAX	I	0	0	1.00
950011	STATE/LOCAL INCOME TAX REFUNDS	I	0	0	1.00
950021	OTHER TAXES	I	0	0	1.00
950022	PERSONAL PROPERTY TAXES	I	0	0	1.00
950023	OTHER TAX REFUNDS	I	0	0	1.00
980000	INCOME BEFORE TAXES	I	0	0	1.00
980010	FAMILY SIZE	I	0	0	1.00
980020	AGE OF REFERENCE PERSON	I	0	0	1.00
980030	NUMBER OF EARNERS	I	0	0	1.00
980040	NUMBER OF VEHICLES	I	0	0	1.00
980050	NUMBER OF PERSONS UNDER 18	I	0	0	1.00
980060	NUMBER OF PERSONS 65 AND OVER	I	0	0	1.00
980070	INCOME AFTER TAXES	I	0	0	1.00
980090	PCT HOMEOWNER	I	0	0	1.00
980210	PCT MALE REF PERSON	I	0	0	1.00
980220	PCT FEMALE REF PERSON	I	0	0	1.00
980230	PCT HOMEOWNER WITH MORT	I	0	0	1.00
980240	PCT HOMEOWNER WITHOUT MORT	I	0	0	1.00
980250	PCT HOMEOWNER MORT NOT RPTED	I	0	0	1.00
980260	PCT RENTER	I	0	0	1.00
980270	PCT BLACK REF PERSON	I	0	0	1.00
980280	PCT NON-BLACK REF PERSON	I	0	0	1.00
980290	PCT REF PERSON W/ ELEM EDUC	I	0	0	1.00
980300	PCT REF PERSON W/ HIGH EDUC	I	0	0	1.00
980310	PCT REF PERSON W/ COLL EDUC	I	0	0	1.00
980320	PCT REF PERSON W/ NO EDUC/ OTH	I	0	0	1.00
980330	PCT VEHICLE OWNER	I	0	0	1.00
980340	PCT CU W/1/MORE LEASED VEHICLES	I	0	0	1.00
980350	PCT CU W/1/MORE OWN OR LEASED VEH	I	0	0	1.00
980360	TOTAL NUM LEASED VEHICLES	I	0	0	1.00
990900	RENT/INST DWASH/HOODS/DISPOSAL	I	1	0	1.00
990910	MAT FOR TER/PST CNT, CAP IM, RNTR	I	1	0	1.00
990920	MAT FOR ADD/REMOD/ETC RNTR	I	1	0	1.00

Taxable Items by UCC

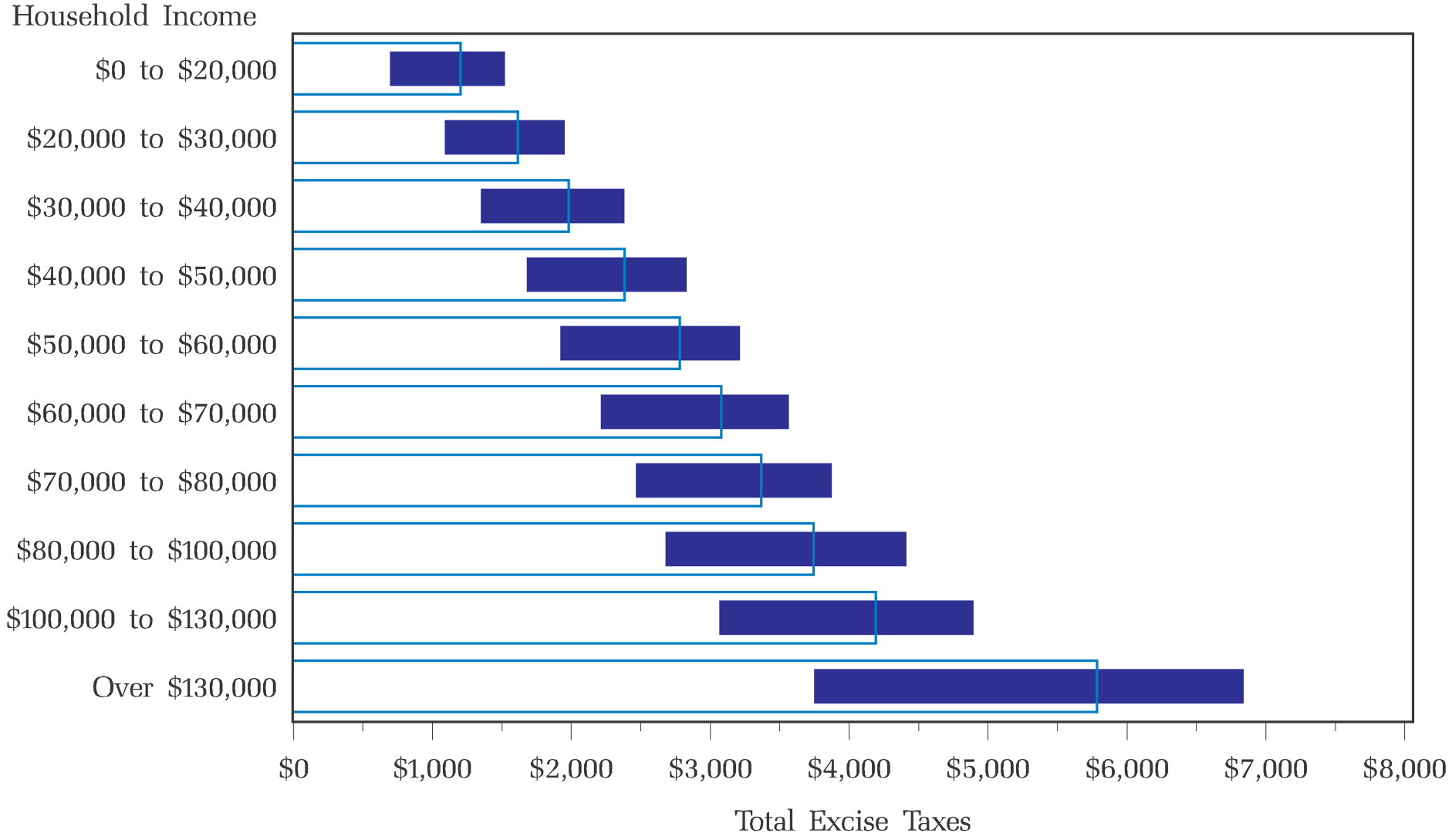
990930	MAT REMOD ETC MAINT/REP OWND	I	1	0	1.00
990940	MAT REMOD ETC MAINT/REP OWNV	I	1	0	1.00
990950	DWEL CONST/ADD-RNTR/UDR CN/SEC HM	I	1	0	1.00

Total Tax Amounts by Imputation Group In Millions

Tax	Group1	Group2	Group3	Group4	Group5	Group6	Group7
Salestax	3,392.3	3,416.9	3,374.3	3,368.7	3,400.1	3,402.5	3,415.2
Liquorsalestax_container	38.2	40.0	42.9	42.2	39.4	40.1	37.2
Liquorvoltax_container	41.1	43.0	46.2	45.4	42.4	43.2	40.1
Liquorsalestax_drink	9.0	9.3	9.1	9.6	9.2	9.6	9.2
Liquorvoltax_drink	15.9	16.5	16.1	16.9	16.3	16.9	16.3
Winelitertax	12.7	13.0	12.1	13.3	12.9	13.2	13.4
Beertax	29.1	31.4	28.7	29.9	30.8	28.7	29.3
Instax	143.2	143.8	141.0	144.5	143.1	143.2	144.4
Cigtax	328.9	340.0	322.0	328.5	324.1	330.5	333.4
Othertobtax	31.0	31.3	32.6	34.3	32.4	35.3	33.1
ElecPUT	63.9	63.9	63.9	63.9	63.4	64.0	63.6
NatgasPUT	13.6	13.7	13.5	13.7	13.8	13.7	13.5
WaterseweragePUT	14.1	13.9	14.1	13.9	13.9	14.1	13.9
GarbagePUT	9.8	10.1	9.9	10.2	10.0	10.0	10.1
IntercityPUT	3.3	3.7	3.5	3.6	3.5	3.8	3.4
IntracityPUT	4.4	4.5	4.3	4.2	4.6	4.5	4.1
Gastax	506.4	505.0	507.7	509.0	501.2	505.3	514.0

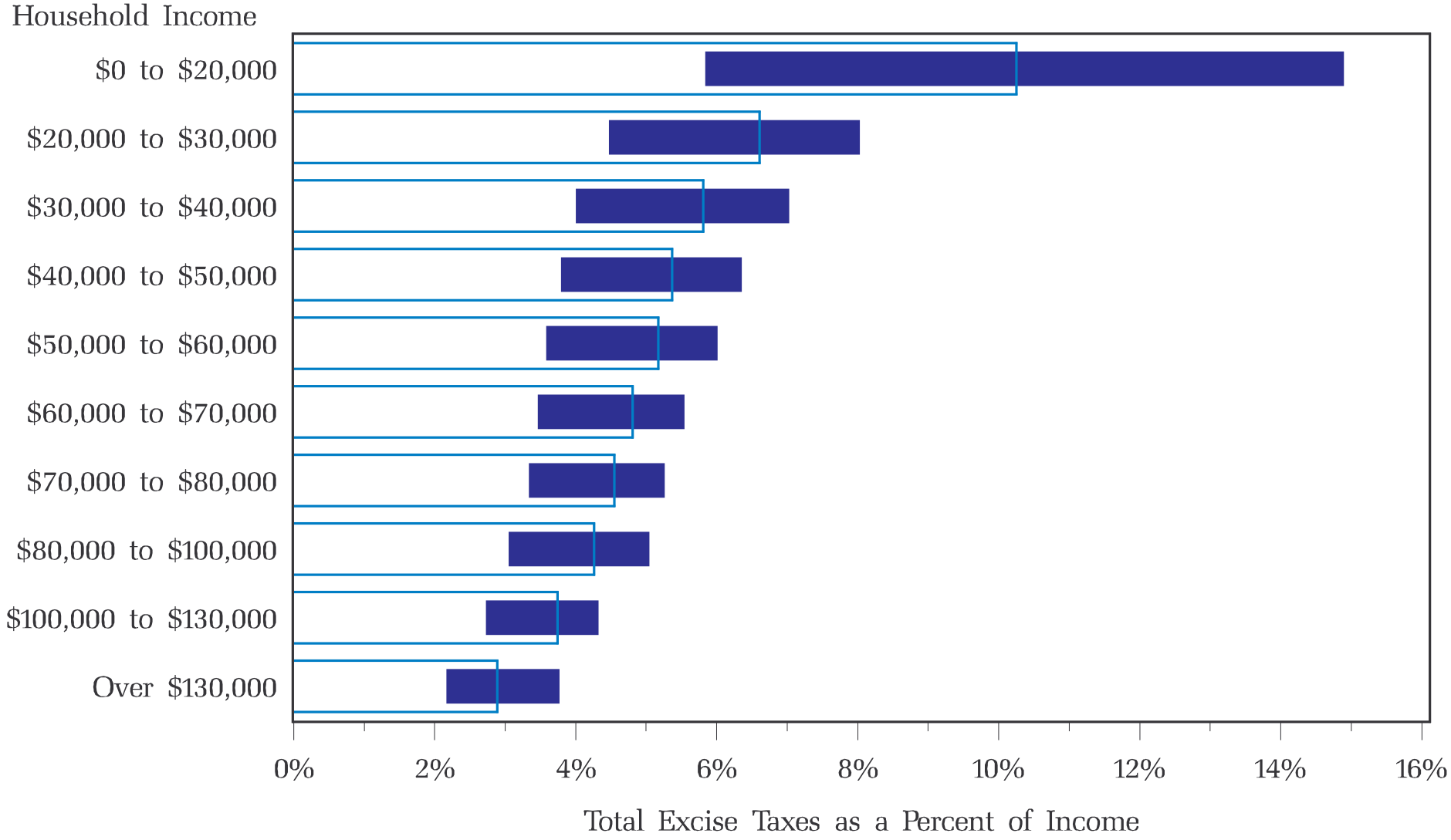
Total Excise Taxes

Average Tax and Interquartile Range



Total Excise Taxes

Average Tax as a Percent of Income and Interquartile Range



Income compared to outlays

Average Outlays by Income Category

Average for all Imputation Groups

1999 HOUSEHOLD TOTAL INCOME	Average Outlays
$0 to $20,000	17,902
$20,000 to $30,000	23,146
$30,000 to $40,000	27,145
$40,000 to $50,000	31,941
$50,000 to $60,000	36,052
$60,000 to $70,000	40,118
$70,000 to $80,000	43,003
$80,000 to $100,000	47,581
$100,000 to $130,000	54,639
Over $130,000	77,599

Income compared to outlays

```
*************************************************************************;
* program:    Total Consumption by SPS ID                                ;
* programmer: Rick Peterson                                              ;
* project:    Washington Excise Tax Microsimulation Model                ;
* date:       March 4, 2002                                              ;
*                                                                        ;
* purpose:    Creates file with income and total consumption by SPSID    ;
*                                                                        ;
*------------------------------------------------------------------------;
* libraries: Altcons - Contains one consumption data set for each        ;
*                      of the 7 imputation groups                        ;
*            extaxmdl - Location of excise tax model data sets           ;
*------------------------------------------------------------------------;
* incoming:   aggi99.txt - contains definition of total consumption      ;
*             altcons.consumption&x - consumption data for each group    ;
*------------------------------------------------------------------------;
* formats:    Formats for excise tax microsimulation model 1.sas         ;
*                                                                        ;
*------------------------------------------------------------------------;
* outgoing:   Totalconsump1 - Totalconsump7 containing total consumption ;
*              for each SPS ID                                           ;
*                                                                        ;
*------------------------------------------------------------------------;
* reports:   Tables of average total consumption by income category      ;
*                                                                        ;
*------------------------------------------------------------------------;
* changes:                                                               ;
*                                                                        ;
*------------------------------------------------------------------------;
* notes:     The definition of total outlays from Rogers and Gray,       ;
*            Monthly Labor Review, Dec 1994 is used. Total Outlays       ;
*            equals CEX total consumption plus principal payments        ;
*            on home mortgages and financed vehicles less purchase       ;
*            price of financed vehicles. However, Rogers and Gray        ;
*            include pensions and social security which are removed      ;
*            here.                                                       ;
*************************************************************************;

*------------------------------------------------------------------------;
*Bring in household data from WAPOP to merge with consumption data;
*------------------------------------------------------------------------;
Data z;
set popsur.sps00f04;
where pnum=1;
keep id fnlwgt hhinc;
run;
proc sort data=z;
by id;
run;
*------------------------------------------------------------------------;
*Bring in file for CEX aggregation scheme;
*------------------------------------------------------------------------;
filename agg "c:\data\interview survey\aggi99.txt";

data agfile;
  infile agg lrecl=20;
  input @3 ucc $6.  @10 gift $1.
       @15 line$6.;
      if gift='2';
      run;
```

Income compared to outlays

```
*-------------------------------------------------------------------------;
*Assign line code for repayment of vehicle loan principal payments;
*This is the same code used for principal payments on property mortgages;
*Data below includes payments for nonfinanced vehicles (net outlays),
      cash downpayment on financed vehicles, and
      principal payments on financed vehicles;
*These expenditures will subsititute for net outlays on vehicles;
*-------------------------------------------------------------------------;
data x;
input ucc $ 1-6 line $ 8-13;
cards;
870101 144500
870102 144500
870104 144500
870201 144500
870202 144500
870204 144500
870301 144500
870302 144500
870304 144500
870401 144500
870402 144500
870404 144500
870501 144500
870502 144500
870504 144500
870605 144500
870606 144500
870608 144500
870701 144500
870702 144500
870704 144500
870801 144500
870802 144500
870804 144500
run;
data agfile2;
set agfile x;
proc sort data=agfile2;
by ucc;
run;
*-------------------------------------------------------------------------;
*Attach CEX agg scheme to summarized consumption data;
*Consumption data from program - Create Alt Consumption Data Set;
*-------------------------------------------------------------------------;
%macro loop;
%do x = 1 %to 7;
Proc SQL noprint;
create table work.consumptionbyline as
select *
  from  altcons.consumption&x, agfile2
  where consumption&x..ucc = agfile2.ucc;
  quit;

*Restrict data to items in total consumption plus principal payments;
*Drop net outlays on vehicles;
*Drop pension and social security contribtions;

  data one;
  set consumptionbyline;
```

Income compared to outlays

```
  where line = '104500' or line='144500';
  if ucc in ('450110', '450210', '450220', '460110', '460901',
             '460902') then delete;
    if ucc in ('800910', '800920', '800931', '800932', '800940')
              then delete;
  run;

  proc sort;
  by  id;
  run;

  proc summary data=one;
  by id;
  var cost;
  output out=two sum=;
  run;

*Merge consumption file with pop file and weight consumption;

data extaxmdl.totalconsbyid&x;
merge two(in=a) z;
by id;
if cost = '.' then cost=0;
wcost = cost*fnlwgt;
if not a then delete;
drop _freq_ _type_ ;
run;
proc sort;
by id;

proc delete data=consumptionbyline;
run;
%end;
%mend Loop;
%Loop;
*-------------------------------------------------------------------------;
*Create table of total consumption by income category by imputation group;
*-------------------------------------------------------------------------;
data x;
merge extaxmdl.totalconsbyid1 (rename= wcost=exp1 drop= cost)
extaxmdl.totalconsbyid2 (rename= wcost=exp2 drop= cost)
extaxmdl.totalconsbyid3 (rename= wcost=exp3 drop= cost)
extaxmdl.totalconsbyid4 (rename= wcost=exp4 drop= cost)
extaxmdl.totalconsbyid5 (rename= wcost=exp5 drop= cost)
extaxmdl.totalconsbyid6 (rename= wcost=exp6 drop= cost)
extaxmdl.totalconsbyid7 (rename= wcost=exp7 drop= cost);
by id;
run;

proc sort data=x;
by hhinc;
run;

proc summary data=x;
by hhinc;
var fnlwgt exp1 exp2 exp3 exp4 exp5 exp6 exp7;
format hhinc incfmtd.;
output out=xx sum=;
run;
```

Income compared to outlays

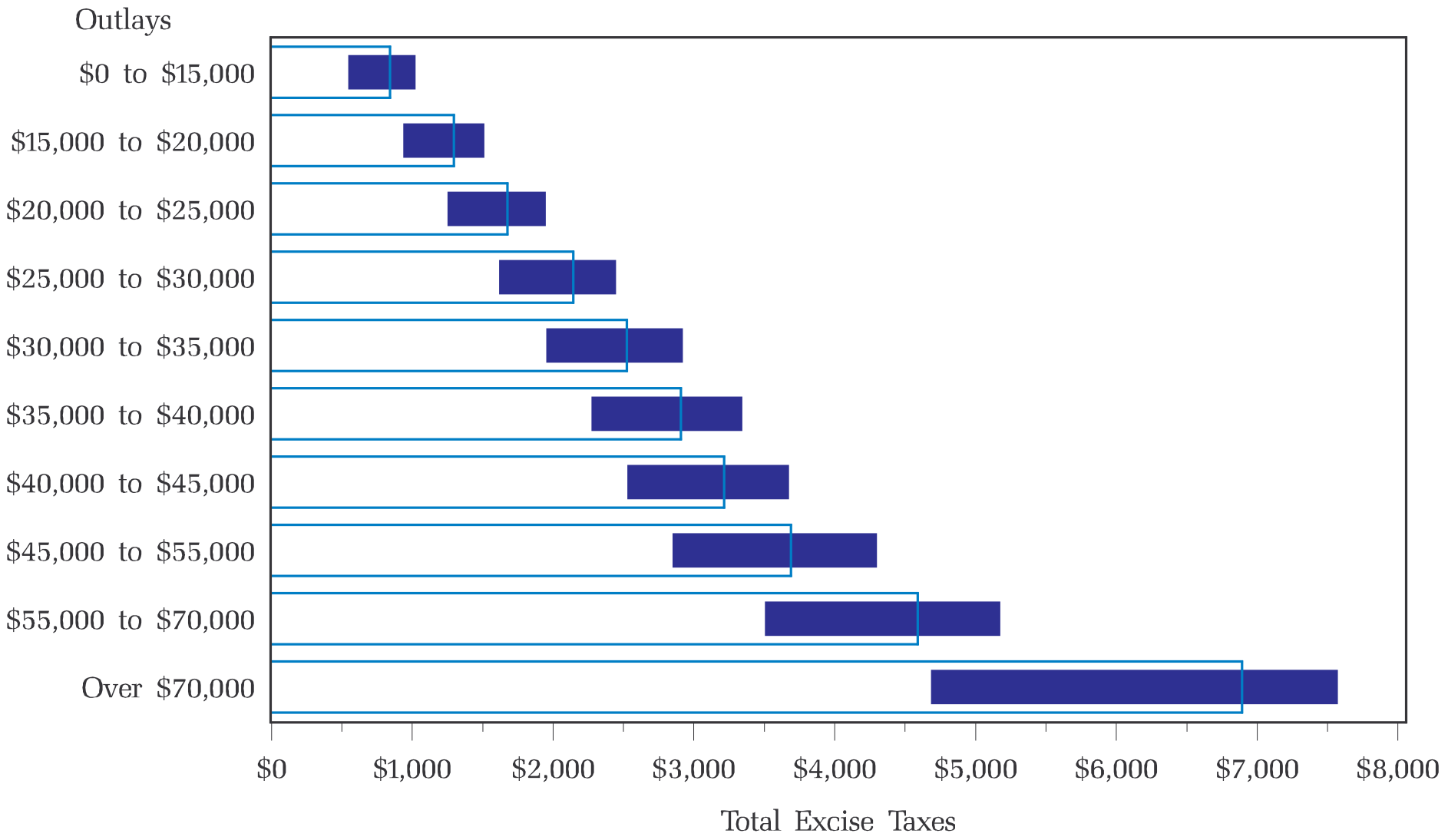
```
data xxx;
set xx;
array aveoutl {7} aveoutl1-aveoutl7;
array expend {7} exp1-exp7;
do i=1 to 7;
aveoutl{i}=expend{i}/fnlwgt;
end;
label aveoutl1 = 'Average Outlay 1'
aveoutl2 = 'Average Outlay 2'
aveoutl3 = 'Average Outlay 3'
aveoutl4 = 'Average Outlay 4'
aveoutl5 = 'Average Outlay 5'
aveoutl6 = 'Average Outlay 6'
aveoutl7 = 'Average Outlay 7';
run;

Title1  h=3 c=black  'Average Outlays by Income Category';
footnote h=1 j=l 'Washington Excise Tax Microsimulation Model'
j=r 'April 15, 2002';
proc print data=xxx noobs labels;
var hhinc aveoutl1-aveoutl7;
format hhinc incfmtd.;
format aveoutl1-aveoutl7 comma10.;
run;

*-------------------------------------------------------------------------;
*Create table of total consumption by income category for all imputation
groups;
*-------------------------------------------------------------------------;
data w;
set extaxmdl.totalconsbyid1 (drop= cost)
extaxmdl.totalconsbyid2 (drop= cost)
extaxmdl.totalconsbyid3 (drop= cost)
extaxmdl.totalconsbyid4 (drop= cost)
extaxmdl.totalconsbyid5 (drop= cost)
extaxmdl.totalconsbyid6 (drop= cost)
extaxmdl.totalconsbyid7 (drop= cost);
run;

proc sort data=w;
by hhinc;
run;

proc summary data=w;
by hhinc;
var fnlwgt wcost;
format hhinc incfmtd.;
output out=ww sum=;
run;

data www;
set ww;
aveoutl=wcost/fnlwgt;
label aveoutl='Average Outlays';
run;

Title1 h=3 c=black  'Average Outlays by Income Category';
Title2 h=2 "Average for all Imputation Groups";
footnote h=1 j=l 'Washington Excise Tax Microsimulation Model'
j=r 'April 15, 2002';
```

Income compared to outlays

```
proc print data=www noobs labels;
var hhinc aveoutl;
format hhinc incfmtd.;
format aveoutl comma10.;
run;
```

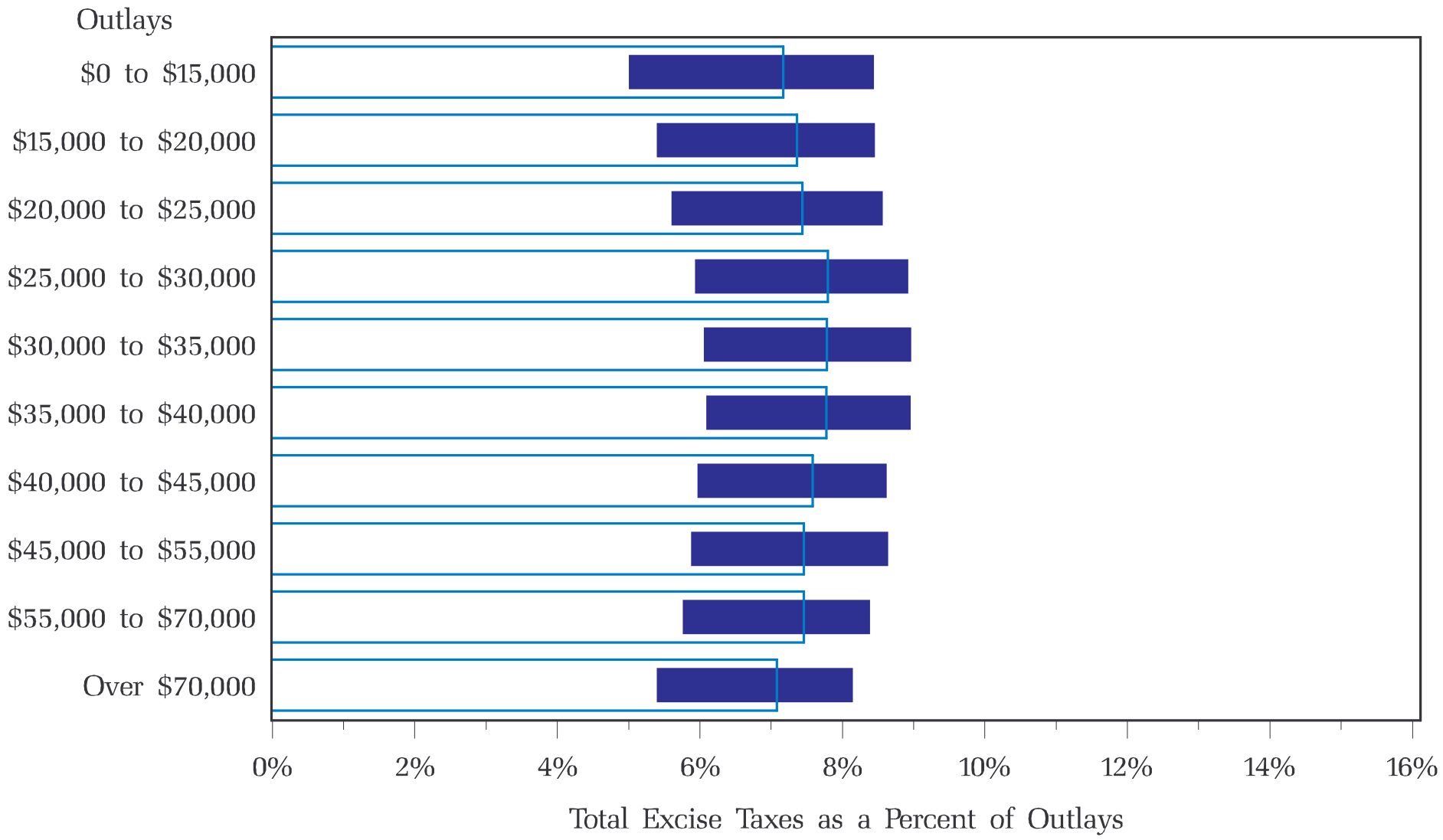
Total Excise Taxes

Average Tax and Interquartile Range



Total Excise Taxes

Average Tax as a Percent of Outlays and Interquartile Range



```
*************************************************************************;
* program:    Create combined consumption tax base                      ;
* programmer: Rick Peterson                                             ;
* project:    Washington Excise Tax Microsimulation Model               ;
* date:       March 27, 2002                                            ;
*                                                                       ;
* purpose:    Creates an excise tax base for each of the seven matched  ;
*             SPS data sets                                             ;
*                                                                       ;
*-----------------------------------------------------------------------;
* libraries:   extaxmdl - location of excise tax model data sets        ;
*                                                                       ;
*-----------------------------------------------------------------------;
* incoming:    expn1-expn4 - Diary survey consumption data               ;
*              mtab1-mtab4,Itab1-Itab4 - Interview survey consumption data;
*              Taxable items list.xls - Excise tax bases by UCC codes   ;
*-----------------------------------------------------------------------;
* formats:    None                                                      ;
*                                                                       ;
*-----------------------------------------------------------------------;
* outgoing:   extaxmdl.taxbase1 - taxbase7  - One data set for each of  ;
*             the seven SPS matched data set                            ;
*                                                                       ;
*-----------------------------------------------------------------------;
* reports:   None                                                       ;
*                                                                       ;
*-----------------------------------------------------------------------;
* changes:                                                              ;
*                                                                       ;
*-----------------------------------------------------------------------;
* notes:     The program first combines consumption from the Diary and  ;
*            Interview survey. Diary data is used for food, and for items ;
*            not included in the Interview survey. Otherwise consumption ;
*            data from the Interview survey is used. The combined       ;
*            consumption data is matched with the external file which   ;
*            identifies UCCs associated with the tax bases for the excise ;
*            taxes.  This file, called taxbase, contains all the taxable ;
*            consumption for each CU in the CEX quarterly data sets.    :
*            This file is then separated into diary and interview       ;
*            data sets by quarter.  Then the data is matched with the SPS ;
*            identification numbers created in hot decking programs.    ;
*            Data is summarized by SPS id number for each of the seven  ;
*            sets of matched data sets.                                 ;
*************************************************************************;
*-----------------------------------------------------------------------;
*Read in consumption data from Diary data sets;
*-----------------------------------------------------------------------;
%let y = 99;
filename expn1 "c:\data\diary survey\expnd&y.1.txt";
filename expn2 "c:\data\diary survey\expnd&y.2.txt";
filename expn3 "c:\data\diary survey\expnd&y.3.txt";
```

```
filename expn4 "c:\data\diary survey\expnd&y.4.txt";
%macro loop;
%do x = 1 %to 4;
data expn&x;
   infile expn&x lrecl=40;
   input @1 newid 8. @35 ucc $6. @10 cost 12.5 ;
   dataset = "D&x";
   run;
%end;
%mend Loop;
%Loop;

*-------------------------------------------------------------------------;
*Combine the diary consumption data sets;
*Keep only selected data from Diary Survey;
*-------------------------------------------------------------------------;

Data Diarycombined;
set expn1 expn2 expn3 expn4;
ucccode=input(ucc,$6.);
if 10110 <=ucccode<=200536 then output;
else if ucccode in (220400, 220410, 270905, 320380, 320430) then output;
else if 320905 <= ucccode <= 330510 then output;
else if 330610 <= ucccode <=340120 then output;
else if ucccode in (340913, 450900, 470114, 480212, 490000, 490316,
520541, 530903, 550210, 550310, 550410, 550900, 570902, 590900,
610220, 610310, 610901, 610902, 610903, 620510, 620610, 620710,
620911, 620913, 630220, 630900, 640110, 640120, 640220, 640310,
640410, 660000, 680903) then output;
run;

*-------------------------------------------------------------------------;
*Read in consumption data from Interview data sets;
*-------------------------------------------------------------------------;

filename mtab1 "c:\data\interview survey\mtabi&y.1x.txt";
filename mtab2 "c:\data\interview survey\mtabi&y.2.txt";
filename mtab3 "c:\data\interview survey\mtabi&y.3.txt";
filename mtab4 "c:\data\interview survey\mtabi&y.4.txt";

%macro loop2;
%do x= 1 %to 4 %by 1;
data mtab&x;
      infile mtab&x lrecl=35;
      input @1 newid 8.
            @9 ucc $6.
            @15 cost 12.4;
            dataset = "I&x";
proc sort;
by newid;
run;
%end;
%mend Loop2;
```

```
%Loop2;

filename itab1 "c:\data\interview survey\itabi&y.1x.txt";
filename itab2 "c:\data\interview survey\itabi&y.2.txt";
filename itab3 "c:\data\interview survey\itabi&y.3.txt";
filename itab4 "c:\data\interview survey\itabi&y.4.txt";

%macro loop3;
%do x= 1 %to 4 %by 1;
data itab&x;
      infile itab&x lrecl=35;
      input @1 newid 8.
            @15 ucc $6.
            @22 cost 12.4;
        dataset = "I&x";
proc sort;
by newid;
run;
%end;
%mend Loop3;
%Loop3;

Data Interviewcombined;
set mtab1 mtab2 mtab3 mtab4
itab1 itab2 itab3 itab4;
by newid;
if ucc='710110' then cost =cost*4;
run;

*-------------------------------------------------------------------------;
*Combine Diary and Interview data;
*-------------------------------------------------------------------------;

data extaxmdl.consumption;
set diarycombined interviewcombined;
run;

proc sort data=extaxmdl.consumption;
by ucc;
run;

*-------------------------------------------------------------------------;
*Delete preliminary data sets;
*-------------------------------------------------------------------------;
proc datasets;
delete expn1 expn2 expn3 expn4 mtab1 mtab2 mtab3 mtab4
itab1 itab2 itab3 itab4 diarycombined interviewcombined;
run;
```

```
*------------------------------------------------------------------------------;
*Import taxable items list from spreadsheet;
*------------------------------------------------------------------------------;

proc import datafile="c:\data\excise tax model\Taxable items list.xls"
out=taxed_items replace;
run;

*Change formats of ucc, survey, and labels;

data taxed_items;
set taxed_items;
attrib ucc length = $6;
ucc=put(ucccode,6.);
if length(left(ucc))=4 then ucc='00'||trim(left(ucc));
if length(left(ucc))=5 then ucc='0'||trim(left(ucc));
drop ucccode surveycode ucccodelabel;
run;


*------------------------------------------------------------------------------;
*Merge consumption data set and taxable items list;
*Calculate tax bases and taxes;
*------------------------------------------------------------------------------;

data extaxmdl.taxbase;
merge extaxmdl.consumption taxed_items;
by ucc;
*Sales tax base;
            Salesbase=Baseadj*current*cost/(1+.084);
            Alt1base=Baseadj*Alt1*cost/(1+.084);
            Alt2base=Baseadj*cost/(1+.084);

*Alcohol tax bases;
      *Liquor sales and liter in container;
      *Base calculated at average price of $10 per 750 ml bottle;
      if other =3 then liquorsalesbase_container = Baseadj*cost/(1+.205);
      else liquorsalesbase_container = 0;
      if other = 3 then liquorvolbase_container = (Baseadj*cost/10)*.750;
      else liquorvolbase_container = 0;

      *Liquor sales and liter by the drink;
      *Base calculated assuming 1.5 oz drinks;
    *which equals an average of 16.8 drinks per 750 ml bottle;
      *at $3.25 per drink;
      if other = 6 then liquorsalesbase_drink = Baseadj*cost/(1+.084);
      else liquorsalesbase_drink = 0;
      if other = 6 then liquorvolbase_drink = ((Baseadj*cost/3.25)/16.8)*.750;
      else liquorvolbase_drink = 0;
```

```
      *Wine tax;
      *Wine sold in containers is based on average
      price of $5.00, including tax, for a 750 ml bottle;
      *Wine by the glass - assumed effective price of $20 per 750 ml bottle;
      if other = 2 then Wineliterbase = (Baseadj*cost/5)*.75;
      Else if other = 5 then Wineliterbase = (Baseadj*cost/20)*.75;
      Else Wineliterbase =0;

      *Beer Tax - rate per 31 gallons;
      *Note - a six pack of 12oz bottles is 72 oz. which is (72/128) gallons;
      *Assume $4 price per six pack for beer sold in containers;
      *Assume $8 price per six pack for beer sold by drink;
      if other = 1 then beerbase = (Baseadj*cost/4)*(72/128)/31;
      Else if other =4 then beerbase = (Baseadj*cost/8)*(72/128)/31;
      Else beerbase = 0;

*Insurance Tax;
      if other = 7 then Insbase = Baseadj*cost/(1+.02);
      Else Insbase = 0;

*Cigarette and other tobacco products Tax;
      *Assume cigarette pack price is $4;
      if other = 8 then cigbase = Baseadj*cost/4;
      else cigbase = 0;

      *Assume wholesale mark up is 50% on other tobacco products;
      if other = 9 then othertobbase = 0.842*(Baseadj*cost/1.5)/(1+.749);
      else othertobbase = 0;

*Real Estate Excise Tax;
      if other = 10 then Reetbase = 10*Baseadj*cost;
      Else reetbase = 0;

*Public Utility Tax;
      if other = 11 then elecbase=Baseadj*cost/(1+.03893);
      else elecbase = 0;
      if other = 12 then naturalgasbase = Baseadj*cost/(1+.03852);
      else naturalgasbase = 0;
      if other = 13 then waterseweragebase =
Baseadj*cost/(1+(.05029+.03852)/2);
      else waterseweragebase = 0;
      if other = 14 then garbagebase = Baseadj*cost/(1+.036);
      else garbagebase = 0;
      if other = 17 then intercitybase = Baseadj*cost/(1+.01926);
      else intercitybase = 0;
      if other = 18 then intracitybase =  Baseadj*cost/(1+.00642);
      else intracitybase = 0;
```

```
*Gas Tax;
      if other = 15 then gasbase = Baseadj*cost/1.4;
      else gasbase = 0 ;
run;

*-------------------------------------------------------------------------;
*Separate data into quarterly Diary and Interview data sets;
*Summarize tax bases by CEX id number;
*-------------------------------------------------------------------------;

%macro loop;
%do x = 1 %to 4;

data a;
set extaxmdl.taxbase;
where dataset = "D&x";
proc summary data=a;
class newid;
var Salesbase Alt1base Alt2base liquorsalesbase_container
liquorvolbase_container
liquorsalesbase_drink liquorvolbase_drink
Wineliterbase beerbase Insbase cigbase othertobbase reetbase elecbase
 naturalgasbase waterseweragebase garbagebase intercitybase intracitybase
 gasbase;
output out=extaxmdl.newidDiary&x sum=;
proc sort data=extaxmdl.newidDiary&x;
by newid;
run;

data a;
set extaxmdl.taxbase;
where dataset="I&x";
proc summary data=a;
class newid;
var Salesbase Alt1base Alt2base liquorsalesbase_container
liquorvolbase_container
liquorsalesbase_drink liquorvolbase_drink
Wineliterbase beerbase Insbase cigbase othertobbase reetbase elecbase
 naturalgasbase waterseweragebase garbagebase intercitybase intracitybase
 gasbase;
output out=extaxmdl.newidInterv&x sum=;
proc sort data=extaxmdl.newidInterv&x;
by newid;
run;

%end;
%mend Loop;
%Loop;
```

```
*-------------------------------------------------------------------------;
*Separate the seven matched SPS data sets into 4 X 2 X 7 data sets. One data
set data for each matched group by quarter by survey type;
*-------------------------------------------------------------------------;

%macro Loop2;
%do x= 1 %to 4;
%do z = 1 %to 7;
Data one;
set mdimpute.dimpute&x;
rename newid&z = newid;
keep id newid&z;
Proc sort data=one  out=D&x&z;
by newid;
run;
Data one;
set miimpute.iimpute&x;
rename newid&z = newid;
keep id newid&z;
Proc sort data=one  out=I&x&z;
by newid;
run;
%end;
%end;
%mend Loop2;
%Loop2;

*-------------------------------------------------------------------------;
*Create 2 (D,I) by 4 (qtrs) by 7 (number of groups) data sets from
consumption data with each WAPOP household associated with summary
expenditure data for the NEWID attached to the WAPOP household;
*-------------------------------------------------------------------------;

%macro Loop3;
%do x = 1 %to 4;
%do z = 1 %to 7;
Proc SQL noprint;
create table work.DA&x&z as
select *
  from  D&x&z , extaxmdl.newidDiary&x
  where D&x&z..newid = newidDiary&x..newid ;
  quit;
Proc sort data=DA&x&z;
by id;
Proc SQL noprint;
create table work.IA&x&z as
select *
  from  I&x&z , extaxmdl.newidinterv&x
  where I&x&z..newid = newidinterv&x..newid ;
```

```
  quit;
Proc sort data=IA&x&z;
by id;
run;
%end;
%end;
%mend Loop3;
%Loop3;

*------------------------------------------------------------------------;
*Combine Diary and Interview data for each SPS match group;
*First combine the 4 qtrs of consumtion data by Diary and Interview
 and sum by SPS id;
*Second, merge in information on the number of qtrs sucessfully matched
 by Diary and Interview and adjust the data accordingly;
*Combine Diary and Interview data and sum by SPS id;
*------------------------------------------------------------------------;

%macro loop4;
%do x=1 %to 7;
data combine_qtrs;
set DA1&x DA2&x DA3&x DA4&x;
by id;

proc summary data=combine_qtrs;
by id;
var Salesbase Alt1base Alt2base liquorsalesbase_container
liquorvolbase_container
liquorsalesbase_drink liquorvolbase_drink
Wineliterbase beerbase Insbase cigbase othertobbase reetbase elecbase
 naturalgasbase waterseweragebase garbagebase intercitybase intracitybase
 gasbase;
output out = sumcombined sum=;

data dcombined&x;
merge sumcombined mdimpute.dqtrsmatched;
by id;
  costadj = 52/dqtrs;
  Salesbase = costadj*Salesbase;
  Alt1base  = costadj*Alt1base;
  Alt2base  = costadj*Alt2base;
  liquorsalesbase_container  = costadj*liquorsalesbase_container;
  liquorvolbase_container  = costadj*liquorvolbase_container;
  liquorsalesbase_drink  = costadj*liquorsalesbase_drink;
  liquorvolbase_drink = costadj*liquorvolbase_drink;
  Wineliterbase  = costadj*Wineliterbase;
  beerbase = costadj*beerbase;
  Insbase  = costadj*Insbase;
  cigbase  = costadj*cigbase;
```

```
  othertobbase  = costadj*othertobbase;
  reetbase  = costadj*reetbase;
  elecbase = costadj*elecbase;
  naturalgasbase  = costadj*naturalgasbase;
  waterseweragebase  = costadj*waterseweragebase;
  garbagebase  = costadj*garbagebase;
  intercitybase  = costadj*intercitybase;
  intracitybase  = costadj*intracitybase;
  gasbase = costadj*gasbase;

*End of Diary processing and begining of Interview processing;

data combine_qtrs;
set IA1&x IA2&x IA3&x IA4&x ;
by id;

proc summary data=combine_qtrs;
by id;
var Salesbase Alt1base Alt2base liquorsalesbase_container
liquorvolbase_container
liquorsalesbase_drink liquorvolbase_drink
Wineliterbase beerbase Insbase cigbase othertobbase reetbase elecbase
 naturalgasbase waterseweragebase garbagebase intercitybase intracitybase
 gasbase;
output out = sumcombined sum=;

data Icombined&X;
merge sumcombined mIimpute.iqtrsmatched;
by id;
costadj = 4/dqtrs;
  Salesbase = costadj*Salesbase;
  Alt1base  = costadj*Alt1base;
  Alt2base  = costadj*Alt2base;
  liquorsalesbase_container  = costadj*liquorsalesbase_container;
  liquorvolbase_container  = costadj*liquorvolbase_container;
  liquorsalesbase_drink  = costadj*liquorsalesbase_drink;
  liquorvolbase_drink = costadj*liquorvolbase_drink;
  Wineliterbase  = costadj*Wineliterbase;
  beerbase = costadj*beerbase;
  Insbase  = costadj*Insbase;
  cigbase  = costadj*cigbase;
  othertobbase  = costadj*othertobbase;
  reetbase  = costadj*reetbase;
  elecbase = costadj*elecbase;
  naturalgasbase  = costadj*naturalgasbase;
  waterseweragebase  = costadj*waterseweragebase;
  garbagebase  = costadj*garbagebase;
  intercitybase  = costadj*intercitybase;
  intracitybase  = costadj*intracitybase;
```

```
  gasbase = costadj*gasbase;

*End of Interview processing;
*Combine adjusted Diary and Interview data and sum by id;

data combined&x;
set dcombined&x icombined&x;
by id;

Proc summary data=combined&x;
by id;
var Salesbase Alt1base Alt2base liquorsalesbase_container
liquorvolbase_container
liquorsalesbase_drink liquorvolbase_drink
Wineliterbase beerbase Insbase cigbase othertobbase reetbase elecbase
 naturalgasbase waterseweragebase garbagebase intercitybase intracitybase
 gasbase;
output out=extaxmdl.taxbase&x sum=;
run;
%end;
%mend loop4;
%loop4;
```

```
*************************************************************************;
* program:    Excise Taxes by Income Category                           ;
* programmer: Rick Peterson                                             ;
* project:    Washington Excise Tax Microsimulation Model               ;
* date:       April 27, 2002                                            ;
*                                                                       ;
* purpose:    Creates tables by income category of excise tax amouts    ;
*             and tax as a percent of income. Incorporates data on      ;
*             property tax and creates tables for total taxes           ;
*                                                                       ;
*-----------------------------------------------------------------------;
* libraries: popsur - location of WA POP survey data                    ;
*            Extaxmdl - location of microsimulation model data sets     ;
*            hmdl2002 - location of Homeowner tax model                 ;
*-----------------------------------------------------------------------;
* incoming:   popsur.sps00f04 - WA POP survey                           ;
*             Extaxmdl.taxbase1 - taxbase7 - Excise tax bases from      ;
*             program called 'Create combined consumption tax base'     ;
*             hmdl2002.HandRptx - Prop tax data from 2002 prop tax model;
*-----------------------------------------------------------------------;
* formats:   'Formats for excise tax microsimulation model 1.sas'       ;
*            'Formats for excise tax microsimulation model 2.sas'       ;
*-----------------------------------------------------------------------;
* outgoing:   None                                          ;
*                                                                       ;
*-----------------------------------------------------------------------;
* reports:   Tables 5, 6 9, and 10 for Excise tax microsimulation model ;
*            documentation
;
*-----------------------------------------------------------------------;
* changes:                                                              ;
*                                                                       ;
*-----------------------------------------------------------------------;
* notes:                                                                ;
*                                                                       ;
*                                                                       ;
*************************************************************************;
*-----------------------------------------------------------------------;
*Get household data from WAPOP to merge with tax data;
*-----------------------------------------------------------------------;
Data z;
set popsur.sps00f04;
where pnum=1;
keep id fnlwgt hhinc;
run;
proc sort data=z;
by id;
run;


*-----------------------------------------------------------------------;
*Apply State and Local tax rates to taxbases;
*-----------------------------------------------------------------------;

%macro loop1;
%do x=1 %to 7;

Data excisetaxes&x;
set Extaxmdl.taxbase&x;
*Sales tax;
```

```
salesrate = .084;
            SalesTax = salesbase*salesrate;
*Alcohol taxes;
      *Liquor sales and liter in container;
      Liquorsalesrate_container = .205;
      Liquorliterrate = 2.4408;
            Liquorsalestax_container =
liquorsalesbase_container*Liquorsalesrate_container;
            Liquorvoltax_container = liquorvolbase_container *
Liquorliterrate;

      *Liquor sales and liter by the drink;
      *Assume 50 percent markup after special sales tax is paid;
      Liquorsalesrate_drink = .137;
            Liquorsalestax_drink =
.15*liquorsalesbase_drink*Liquorsalesrate_drink;
            Liquorvoltax_drink = liquorvolbase_drink * Liquorliterrate;

      *Wine tax;
      Wineliterrate = .2292;
            Winelitertax = Wineliterbase*wineliterrate;

      *Beer Tax - rate per 31 gallons;
      Beertaxrate = 8.08;;
            Beertax = beerbase * beertaxrate;

*Insurance Tax;
      Insrate = .02;
            InsTax = insrate*insbase;

*Cigarette and other tobacco products taxes;
      Cigrate = 1.425;
            Cigtax = cigbase*cigrate;
      Othertobrate = 1.294;
            Othertobtax = othertobbase*othertobrate;

*Public Utility Tax;
      ElecPUTrate = .0378+.03201;
            ElecPUT = ElecPUTrate*elecbase;
      NatgasPUTrate = .03852+.03046;
            NatgasPUT = NatgasPUTrate*naturalgasbase;
      WaterseweragePUTrate = ((.05029+.03852)/2)+((.0535+.17192)/2);
            WaterseweragePUT = WaterseweragePUTrate*watersewerageBase;
      GarbagePUTrate = .036+.06372;
            GarbagePUT = GarbagePUTrate*garbagebase;
      IntercityPUTrate = .01926;
            IntercityPUT = IntercityPUTrate*intercitybase;
      IntracityPUTrate = .01926;
            IntracityPUT = IntracityPUTrate*intracitybase;

*Gas Tax;
      Gasrate = .23;
            Gastax = gasrate*gasbase;

AlcoholTaxes = liquorsalestax_container + liquorvoltax_container +
                        liquorsalestax_drink + liquorvoltax_drink +
Winelitertax + Beertax;
TobaccoTaxes = cigtax + othertobtax;
UtilityTaxes =  ElecPUT + NatgasPUT + WaterseweragePUT + GarbagePUT +
                        IntercityPUT + IntracityPUT;
```

```
TotalExciseTaxes = Salestax + Alcoholtaxes + Instax + Tobaccotaxes +
UtilityTaxes + Gastax;
run;

*Merge aggregated tax with household weight from WAPOP;

data excisetaxmerged&x;
merge excisetaxes&x(in=a) z;
by id;
if a;
income=hhinc;
households=1;
drop _freq_ _type_;
run;
%end;
%mend loop1;
%loop1;
*-------------------------------------------------------------------------;
*Calculate total tax and tax as percent of income
by income group for each imputation group;
*-------------------------------------------------------------------------;

*First, calculate tax by income group;

OPTIONS ls=130 ps=50 pageno=1;
%macro loop3;
%do z=1 %to 7;
proc summary nway data=excisetaxmerged&z;
class hhinc ;
format hhinc incfmtd.;
weight fnlwgt;
var TotalExciseTaxes Alcoholtaxes
Tobaccotaxes UtilityTaxes salestax Instax Gastax households income;
output out=weightedtaxbyinc&z sum=;
run;

*Divide by number of households by income group and calc avg tax;

data avgtax;
set weightedtaxbyinc&z;
TotalExciseTaxes = TotalExciseTaxes/households;
Salestax = salestax/households;
Alcoholtaxes = Alcoholtaxes/households;
Instax = instax/households;
Tobaccotaxes = Tobaccotaxes/households;
UtilityTaxes = UtilityTaxes/households;
Gastax = gastax/households;
drop _freq_ _type_;
run;

title1 "Excise Tax for Imputation Group&z" ;

Proc tabulate data = avgtax;
class hhinc;
format hhinc incfmtd.;
var TotalExciseTaxes Salestax
Alcoholtaxes Instax Tobaccotaxes
UtilityTaxes Gastax;
table hhinc,(TotalExciseTaxes Salestax Alcoholtaxes Instax Tobaccotaxes
UtilityTaxes Gastax)*f=comma10.;
```

```
run;

*Divide by total income by income group and calc tax as percent of income;

data avgtaxpercent;
set weightedtaxbyinc&z;
TotalExciseTaxes = TotalExciseTaxes/income;
Salestax = salestax/income;
Alcoholtaxes = Alcoholtaxes/income;
Instax = instax/income;
Tobaccotaxes = Tobaccotaxes/income;
UtilityTaxes = UtilityTaxes/income;
Gastax = gastax/income;
drop _freq_ _type_;
run;

title1 "Excise Tax as Percent of Income for Imputation Group&z" ;

Proc tabulate data = avgtaxpercent ;
class hhinc;
format hhinc incfmtd.;
var TotalExciseTaxes Salestax
Alcoholtaxes Instax Tobaccotaxes
UtilityTaxes Gastax;
table hhinc,(TotalExciseTaxes
Salestax Alcoholtaxes Instax Tobaccotaxes
UtilityTaxes Gastax)*f=percent10.2;
run;
%end;
%mend loop3;
%loop3;

*-------------------------------------------------------------------------;
*Calculate Tax as a percent of income by income group for average of
all seven imputation groups;
*-------------------------------------------------------------------------;

Data two;
set excisetaxmerged1 excisetaxmerged2 excisetaxmerged3 excisetaxmerged4
excisetaxmerged5 excisetaxmerged6 excisetaxmerged7;
run;
Proc summary data = two nway;
class hhinc;
format hhinc incfmtd.;
var TotalExciseTaxes Alcoholtaxes
Tobaccotaxes UtilityTaxes
salestax Instax Gastax households income;
weight fnlwgt;
output out=Aveweightedtaxbyinc sum=;
run;

*Divide by total number of households by income group and calc avg tax;

data avgtax1;
set aveweightedtaxbyinc;
TotalExciseTaxes = TotalExciseTaxes/households;
salestax = salestax/households;
Alcoholtaxes = Alcoholtaxes/households;
Instax = instax/households;
Tobaccotaxes = Tobaccotaxes/households;
```

```
UtilityTaxes = UtilityTaxes/households;
Gastax = gastax/households;
drop _freq_ _type_;
run;

Title1 'Table 5';
title2 "State and Local Excise Tax by Income Category";
title3 "Average of all Imputation Groups" ;

Proc tabulate data = avgtax1 ;
class hhinc;
format hhinc incfmtd.;
var  TotalExciseTaxes
salestax Alcoholtaxes Instax Tobaccotaxes
 UtilityTaxes Gastax;
table hhinc,(TotalExciseTaxes Salestax
Alcoholtaxes Instax Tobaccotaxes
UtilityTaxes Gastax)*f=comma10.;
run;

*Divide by total income by income group and calc tax as percent of income;

data avgtax2;
set aveweightedtaxbyinc;
TotalExciseTaxes = TotalExciseTaxes/income;
Salestax = salestax/income;
Alcoholtaxes = Alcoholtaxes/income;
Instax = instax/income;
Tobaccotaxes = Tobaccotaxes/income;
UtilityTaxes = UtilityTaxes/income;
Gastax = gastax/income;
drop _freq_ _type_;
run;

Title1 "Table 6";
title2 "State and Local Excise Tax as Percent of Income";
title3 "Average of all Imputation Groups" ;

Proc tabulate data = avgtax2;
class hhinc;
format hhinc incfmtd.;
var TotalExciseTaxes salestax
Alcoholtaxes Instax Tobaccotaxes
UtilityTaxes Gastax;
table hhinc,(TotalExciseTaxes salestax
Alcoholtaxes Instax Tobaccotaxes
UtilityTaxes Gastax)*f=percent10.2;
run;

*------------------------------------------------------------------------;
*Create Tables with Excise and Property Taxes;
*------------------------------------------------------------------------;

*Incorporate information on homeowner and renter prop tax;
data x;
set hmdl2002.HandRptx;
propincome=hhinc;
households=1;
keep hhinc proptax propincome households fnlwgt;
run;
```

```
Proc summary nway data=x;
class hhinc;
format hhinc incfmtd.;
var proptax propincome households;
weight fnlwgt;
output out=xx sum=;
run;

*Combine excise and property tax data for average tax table;

data avgtax3;
merge avgtax1(drop=households) xx;
by hhinc;
Proptax = proptax/households;
TotalExciseandPropTaxes = TotalExciseTaxes + Proptax;
run;

Title1 'Table 9';
title2 "State and Local Excise and Property Tax as Percent of Income";
title3 "Average of all Imputation Group";

Proc tabulate data = avgtax3 ;
class hhinc;
format hhinc incfmtd.;
var TotalExciseandPropTaxes TotalExciseTaxes Proptax;
table hhinc,(TotalExciseandPropTaxes TotalExciseTaxes Proptax)*f=comma10.;
run;

*Combine excise and property tax data for tax as percent of income table;

data avgtax4;
merge avgtax2 xx;
by hhinc;
Proptax = proptax/propincome;
TotalExciseandPropTaxes = TotalExciseTaxes + Proptax;
run;

Title1 "Table 10";
title2 "State and Local Excise and Property Tax as Percent of Income";
title3 "Average of all Imputation Group";

Proc tabulate data = avgtax4;
class hhinc;
format hhinc incfmtd.;
var TotalExciseandPropTaxes TotalExciseTaxes Proptax;
table hhinc,(TotalExciseandPropTaxes TotalExciseTaxes
Proptax)*f=percent10.2;
run;
```

```
*************************************************************************;
* program:    Tax Calculator for Microsimulation Model                  ;
* programmer: Rick Peterson                                             ;
* project:    Washington Excise Tax Microsimulation Model               ;
* date:       Nov 21, 2002                                              ;
*                                                                       ;
* purpose:    Calculates excise taxes with current law rates and        ;
*             alternative rates                                         ;
*                                                                       ;
*-----------------------------------------------------------------------;
* libraries:   extaxmdl - location of excise tax model data sets        ;
*              popsur - location of Washington Population Survey data   ;
*-----------------------------------------------------------------------;
* incoming:    Extaxmdl.taxbase&x Tax base for each imputation group    ;
*              popsur.sps00f04 Washington Population Survey             ;
*-----------------------------------------------------------------------;
* formats:     'Formats for microsimulation model 1'                    ;
*              'Formats for microsimulation model 2'                    ;
*-----------------------------------------------------------------------;
* outgoing:   None                                                      ;
*-----------------------------------------------------------------------;
* reports:    Total tax by tax type                                     ;
*             Average tax by income group                               ;
*             Tax as a percent of Income by income group                ;
*-----------------------------------------------------------------------;
* changes:                                                              ;
*                                                                       ;
*-----------------------------------------------------------------------;
* notes:
*************************************************************************;

*-----------------------------------------------------------------------;
*Get household data from WAPOP to merge with tax data;
*-----------------------------------------------------------------------;

Data z;
set popsur.sps00f04;
where pnum=1;
keep id fnlwgt hhinc peopl hhearn99
nwageinc chldrn20 adults21 age hhtype q4p4g;
run;
proc sort data=z;
by id;
run;

*Sum income by income class;

proc summary nway data=z;
class hhinc;
format hhinc incfmtd.;
var hhinc;
weight fnlwgt;
output out=incomebyclass sum=income;
run;

*-----------------------------------------------------------------------;
*Apply tax rates to taxbase for each imputation group;
*-----------------------------------------------------------------------;
```

```
%macro loop1;
%do x=1 %to 7;

Data excisetaxes&x;
set Extaxmdl.taxbase&x;

*Sales tax;
salesrate = .084;
            salestax = salesbase*salesrate;
            alt1tax = .01* alt1base;
            alt2tax = .01* alt2base;

*Alcohol taxes;
      *Liquor sales and liter in container;
      Liquorsalesrate_container = .205;
      AltLiquorsalesrate_container = .205;
      Liquorliterrate = 2.4408;
      AltLiquorliterrate = 2.4408;
            liquorsalestax_container =
liquorsalesbase_container*Liquorsalesrate_container;
            liquorvoltax_container = liquorvolbase_container *
Liquorliterrate;
            Altliquorsalestax_container =
liquorsalesbase_container*AltLiquorsalesrate_container;
            Altliquorvoltax_container = liquorvolbase_container *
AltLiquorliterrate;

      *Liquor sales and liter by drink;
      Liquorsalesrate_drink = .137;
      AltLiquorsalesrate_drink = .137;
            liquorsalestax_drink =
.15*liquorsalesbase_drink*Liquorsalesrate_drink;
            liquorvoltax_drink = liquorvolbase_drink * Liquorliterrate;
            Altliquorsalestax_drink =
.15*liquorsalesbase_drink*AltLiquorsalesrate_drink;
            Altliquorvoltax_drink = liquorvolbase_drink *
AltLiquorliterrate;

      *Wine tax;
      Wineliterrate = .2292;
      AltWineliterrate = .2292;
            Winelitertax = Wineliterbase*wineliterrate;
            AltWinelitertax = Wineliterbase*altwineliterrate;

      *Beer Tax - rate per 31 gallons;
      Beertaxrate = 8.08;
      AltBeertaxrate = 8.08;
            Beertax = beerbase * beertaxrate;
            AltBeertax = beerbase * altbeertaxrate;

*Insurance Tax;
      Insrate = .02;
      Altinsrate = .02;
            Instax = insrate*insbase;
            AltInstax = altinsrate*insbase;

*Cigarette and other tobacco products taxes;
      Cigrate = 1.425;
      Altcigrate = 1.425;
            cigtax = cigbase*cigrate;
```

```
            Altcigtax = cigbase*altcigrate;
      Othertobrate = 1.294;
      Altothertobrate = 1.294;
            othertobtax = othertobbase*othertobrate;
            altothertobtax = othertobbase*altothertobrate;

*Public Utility Tax;
      ElecPUTrate = .0378+.03201;
      AltElecPUTrate = .0378+.03201;
            ElecPUT = ElecPUTrate*elecbase;
            AltElecPUT = AltElecPUTrate*elecbase;
      NatgasPUTrate = .03852+.03046;
      AltNatgasPUTrate = .03852+.03046;
            NatgasPUT = NatgasPUTrate*naturalgasbase;
            AltNatgasPUT = AltNatgasPUTrate*naturalgasbase;
      WaterseweragePUTrate = (.05029+.03852)/2+((.0535+.17192)/2);
      AltWaterseweragePUTrate = (.05029+.03852)/2+((.0535+.17192)/2);
            WaterseweragePUT = WaterseweragePUTrate*waterseweragebase;
            AltWaterseweragePUT = AltWaterseweragePUTrate*waterseweragebase;
      GarbagePUTrate = .036+.06372;
      AltGarbagePUTrate = .036+.06372;
            GarbagePUT = GarbagePUTrate*garbagebase;
            AltGarbagePUT = AltGarbagePUTrate*garbagebase;
      IntercityPUTrate = .01926;
      AltIntercityPUTrate = .01926;
            IntercityPUT = IntercityPUTrate*intercitybase;
            AltIntercityPUT = AltIntercityPUTrate*intercitybase;
      IntracityPUTrate = .01926;
      AltIntracityPUTrate = .01926;
            IntracityPUT = IntracityPUTrate*intracitybase;
            AltIntracityPUT = AltIntracityPUTrate*intracitybase;

*Gas Tax;
      Gasrate = .23;
      Altgastax = .23;
            Gastax = gasrate*gasbase;
            Altgastax = altgastax*gasbase;

Alcoholtaxes = liquorsalestax_container + liquorvoltax_container +
                        liquorsalestax_drink + liquorvoltax_drink +
Winelitertax + Beertax;
AltAlcoholtaxes = Altliquorsalestax_container + Altliquorvoltax_container +
                        Altliquorsalestax_drink + Altliquorvoltax_drink +
AltWinelitertax + AltBeertax;
Tobaccotaxes = cigtax + othertobtax;
AltTobaccotaxes = Altcigtax + altothertobtax;
UtilityTaxes =  ElecPUT + NatgasPUT + WaterseweragePUT + GarbagePUT +
                        IntercityPUT + IntracityPUT;
AltUtilityTaxes = AltElecPUT + AltNatgasPUT + AltWaterseweragePUT +
                        AltGarbagePUT + AltIntercityPUT + AltIntracityPUT;

TotalExciseTaxes = salestax + Alcoholtaxes + Instax + Tobaccotaxes +
UtilityTaxes + Gastax;
AltTotalExciseTaxes = alt1tax + Altalcoholtaxes + AltInstax +
AltTobaccotaxes +
AltUtilityTaxes + Altgastax;
run;
```

```
*------------------------------------------------------------------------------;
*Merge tax amounts with household characteristics from WAPOP;
*------------------------------------------------------------------------------;

data excisetaxmerged&x;
merge excisetaxes&x z ;
by id;
households=1;
income=hhinc;
drop _freq_ _type_;
run;
%end;
%mend loop1;
%loop1;

OPTIONS ls=123 ps=52 pageno=1;

*------------------------------------------------------------------------------;
*Calculate total tax by tax type;
*------------------------------------------------------------------------------;

%macro loop2;
%do z=1 %to 7;
proc summary  data=excisetaxmerged&z;
weight fnlwgt;
var salestax alt1tax alt2tax liquorsalestax_container liquorvoltax_container
Altliquorsalestax_container Altliquorvoltax_container liquorsalestax_drink
liquorvoltax_drink Altliquorsalestax_drink Altliquorvoltax_drink
Winelitertax AltWinelitertax Beertax AltBeertax Instax AltInstax
cigtax Altcigtax othertobtax altothertobtax
ElecPUT AltElecPUT NatgasPUT AltNatgasPUT WaterseweragePUT
AltWaterseweragePUT
GarbagePUT AltGarbagePUT IntercityPUT AltIntercityPUT IntracityPUT
AltIntracityPUT
Gastax Altgastax households peopl;
output out=weightedtax&z sum=;
run;
%end;
%mend loop2;
%loop2;

*Average the data from the seven imputation groups;

Data one;
set weightedtax1 weightedtax2 weightedtax3 weightedtax4
weightedtax5 weightedtax6 weightedtax7;
proc summary  data=one;
var salestax alt1tax alt2tax liquorsalestax_container liquorvoltax_container
Altliquorsalestax_container Altliquorvoltax_container liquorsalestax_drink
liquorvoltax_drink Altliquorsalestax_drink Altliquorvoltax_drink
Winelitertax AltWinelitertax Beertax AltBeertax Instax AltInstax
cigtax Altcigtax othertobtax altothertobtax
ElecPUT AltElecPUT NatgasPUT AltNatgasPUT WaterseweragePUT
AltWaterseweragePUT
GarbagePUT AltGarbagePUT IntercityPUT AltIntercityPUT IntracityPUT
AltIntracityPUT
Gastax Altgastax households peopl;
output out=Aveweightedtax mean=;
run;
```

```
Title1 "Total tax amounts for all groups (Average)";
proc print data=Aveweightedtax;
var salestax alt1tax alt2tax liquorsalestax_container liquorvoltax_container
Altliquorsalestax_container Altliquorvoltax_container liquorsalestax_drink
liquorvoltax_drink Altliquorsalestax_drink Altliquorvoltax_drink
Winelitertax AltWinelitertax Beertax AltBeertax Instax AltInstax
cigtax Altcigtax othertobtax altothertobtax
ElecPUT AltElecPUT NatgasPUT AltNatgasPUT WaterseweragePUT
AltWaterseweragePUT
GarbagePUT AltGarbagePUT IntercityPUT AltIntercityPUT IntracityPUT
AltIntracityPUT
Gastax Altgastax;
format salestax alt1tax alt2tax liquorsalestax_container
liquorvoltax_container
Altliquorsalestax_container Altliquorvoltax_container liquorsalestax_drink
liquorvoltax_drink Altliquorsalestax_drink Altliquorvoltax_drink
Winelitertax AltWinelitertax Beertax AltBeertax Instax AltInstax
cigtax Altcigtax othertobtax altothertobtax
ElecPUT AltElecPUT NatgasPUT AltNatgasPUT WaterseweragePUT
AltWaterseweragePUT
GarbagePUT AltGarbagePUT IntercityPUT AltIntercityPUT IntracityPUT
AltIntracityPUT
Gastax Altgastax comma15.;
run;

*------------------------------------------------------------------------;
*Calculate Tax by income group for each imputation group;
*------------------------------------------------------------------------;

OPTIONS ls=130 ps=50 pageno=1;
%macro loop3;
%do z=1 %to 7;
proc summary nway data=excisetaxmerged&z;
class hhinc ;
format hhinc incfmtd.;
weight fnlwgt;
var TotalExciseTaxes AltTotalExciseTaxes Alcoholtaxes AltAlcoholtaxes
Tobaccotaxes
AltTobaccotaxes UtilityTaxes AltUtilityTaxes
salestax alt1tax alt2tax liquorsalestax_container liquorvoltax_container
Altliquorsalestax_container Altliquorvoltax_container liquorsalestax_drink
liquorvoltax_drink Altliquorsalestax_drink Altliquorvoltax_drink
Winelitertax AltWinelitertax Beertax AltBeertax Instax AltInstax
cigtax Altcigtax othertobtax altothertobtax
ElecPUT AltElecPUT NatgasPUT AltNatgasPUT WaterseweragePUT
AltWaterseweragePUT
GarbagePUT AltGarbagePUT IntercityPUT AltIntercityPUT IntracityPUT
AltIntracityPUT
Gastax Altgastax households peopl;
output out=weightedtaxbyinc&z sum=;
run;
%end;
%mend loop3;
%loop3;

*Add tax from each group and take average;

Data two;
set weightedtaxbyinc1 weightedtaxbyinc2 weightedtaxbyinc3
weightedtaxbyinc4 weightedtaxbyinc5 weightedtaxbyinc6
```

```
weightedtaxbyinc7;
run;
Proc summary data = two nway;
class hhinc;
format hhinc incfmtd.;
var TotalExciseTaxes AltTotalExciseTaxes Alcoholtaxes AltAlcoholtaxes
Tobaccotaxes
AltTobaccotaxes UtilityTaxes AltUtilityTaxes
salestax alt1tax alt2tax liquorsalestax_container liquorvoltax_container
Altliquorsalestax_container Altliquorvoltax_container liquorsalestax_drink
liquorvoltax_drink Altliquorsalestax_drink Altliquorvoltax_drink
Winelitertax AltWinelitertax Beertax AltBeertax Instax AltInstax
cigtax Altcigtax othertobtax altothertobtax
ElecPUT AltElecPUT NatgasPUT AltNatgasPUT WaterseweragePUT
AltWaterseweragePUT
GarbagePUT AltGarbagePUT IntercityPUT AltIntercityPUT IntracityPUT
AltIntracityPUT
Gastax Altgastax households peopl;
output out=Aveweightedtaxbyinc mean=;
run;

*Divide by number of households by income group
and calc avg tax per household;

data avgtax1;
set Aveweightedtaxbyinc;
AvgTotalExciseTaxes = TotalExciseTaxes/households;
AvgAltTotalExciseTaxes = AltTotalExciseTaxes/households;
Avgsalestax = salestax/households;
Avgalt1tax = alt1tax/households;
Avgalt2tax = alt2tax/households;
avgAlcoholtaxes = Alcoholtaxes/households;
avgAltAlcoholtaxes = AltAlcoholtaxes/households;
avgInstax = instax/households;
avgAltinstax = altinstax/households;
avgTobaccotaxes = Tobaccotaxes/households;
avgAltTobaccotaxes = AltTobaccotaxes/households;
avgUtilityTaxes = UtilityTaxes/households;
avgAltUtilityTaxes = AltUtilityTaxes/households;
avgGastax = gastax/households;
avgAltgastax = altgastax/households;
avgpeopl = peopl/households;
drop _freq_ _type_;
run;

title1 "Average Tax for all Groups (Average)" ;
Proc tabulate data = avgtax1;
class hhinc;
format hhinc incfmtd.;
var AvgTotalExciseTaxes AvgAltTotalExciseTaxes Avgsalestax Avgalt1tax
Avgalt2tax
avgAlcoholtaxes avgAltAlcoholtaxes avgInstax avgAltinstax
avgTobaccotaxes avgAltTobaccotaxes
avgUtilityTaxes avgAltUtilityTaxes avgGastax avgAltgastax
avgpeopl;
table hhinc,(AvgTotalExciseTaxes AvgAltTotalExciseTaxes Avgsalestax
Avgalt1tax Avgalt2tax
avgAlcoholtaxes avgAltAlcoholtaxes avgInstax avgAltinstax
avgTobaccotaxes avgAltTobaccotaxes
avgUtilityTaxes avgAltUtilityTaxes avgGastax avgAltgastax)*f=comma10.
```

```
avgpeopl*f=comma10.2;
run;

*Merge in total income by income group
and calc avg tax as percent of income;

data avgtax2;
merge aveweightedtaxbyinc incomebyclass;
by hhinc;
AvgTotalExciseTaxes = TotalExciseTaxes/income;
AvgAltTotalExciseTaxes = AltTotalExciseTaxes/income;
Avgsalestax = salestax/income;
Avgalt1tax = alt1tax/income;
Avgalt2tax = alt2tax/income;
avgAlcoholtaxes = Alcoholtaxes/income;
avgAltAlcoholtaxes = AltAlcoholtaxes/income;
avgInstax = instax/income;
avgAltinstax = altinstax/income;
avgTobaccotaxes = Tobaccotaxes/income;
avgAltTobaccotaxes = AltTobaccotaxes/income;
avgUtilityTaxes = UtilityTaxes/income;
avgAltUtilityTaxes = AltUtilityTaxes/income;
avgGastax = gastax/income;
avgAltgastax = altgastax/income;
drop _freq_ _type_;
run;


title1 "Tax as Percent of Income for all Groups (Average)" ;
Proc tabulate data = avgtax2;
class hhinc;
format hhinc incfmtd.;
var AvgTotalExciseTaxes AvgAltTotalExciseTaxes Avgsalestax Avgalt1tax
Avgalt2tax
avgAlcoholtaxes avgAltAlcoholtaxes avgInstax avgAltinstax
avgTobaccotaxes avgAltTobaccotaxes
avgUtilityTaxes avgAltUtilityTaxes avgGastax avgAltgastax;
table hhinc,(AvgTotalExciseTaxes AvgAltTotalExciseTaxes Avgsalestax
Avgalt1tax Avgalt2tax
avgAlcoholtaxes avgAltAlcoholtaxes avgInstax avgAltinstax
avgTobaccotaxes avgAltTobaccotaxes
avgUtilityTaxes avgAltUtilityTaxes avgGastax avgAltgastax)*f=percent10.2;
run;
```

```
*************************************************************************;
* program:    Tax Calculator for SimTax Model                           ;
* programmer: Rick Peterson                                             ;
* project:    Washington Excise Tax Microsimulation Model               ;
* date:       Nov 21, 2002                                              ;
*                                                                       ;
* purpose:    Calculates excise taxes and income tax alternatives for   ;
*             input into SimTax.  SimTax is spreadsheet model which     ;
*             allows the users to redesign the Washington Tax System    ;
*-----------------------------------------------------------------------;
* libraries:   extaxmdl - location of excise tax model data sets        ;
*              popsur - location of Washington Population Survey data   ;
*-----------------------------------------------------------------------;
* incoming:    Extaxmdl.taxbase&x Tax base for each imputation group    ;
*              popsur.sps00f04 Washington Population Survey             ;
*              extaxmdl.secondwage - data on relationship between first ;
*              and second earner wages.                                 ;
*-----------------------------------------------------------------------;
* formats:     'Formats for microsimulation model 1'                    ;
*              'Formats for microsimulation model 2'                    ;
*-----------------------------------------------------------------------;
* outgoing:   None                                                      ;
*-----------------------------------------------------------------------;
* reports:    Total tax by tax type                                     ;
*             Average tax by income group                               ;
*             Tax as a percent of income by income group                ;
*-----------------------------------------------------------------------;
* changes:                                                              ;
*                                                                       ;
*-----------------------------------------------------------------------;
* notes:
*************************************************************************;

*-----------------------------------------------------------------------;
*Get household data from WAPOP to merge with tax data;
*-----------------------------------------------------------------------;
Data z;
set popsur.sps00f04;
where pnum=1;
keep id fnlwgt hhinc peopl hhearn99
nwageinc chldrn20 adults21 age hhtype q4p4g;
run;
proc sort data=z;
by id;
run;

*Sum income by income class;

proc summary nway data=z;
class hhinc;
format hhinc incfmtd.;
var hhinc;
weight fnlwgt;
output out=incomebyclass sum=income;
run;

*-----------------------------------------------------------------------;
```

```
*Apply tax rates to taxbase for each imputation group;
*-------------------------------------------------------------------------;
%macro loop1;
%do x=1 %to 7;

Data excisetaxes&x;
set Extaxmdl.taxbase&x;


*Sales tax;
salesrate = .084;
            salestax = salesbase*salesrate;
            alt1tax = .01* alt1base;
            alt2tax = .01* alt2base;

*Alcohol taxes;
      *Liquor sales and liter in container;
      Liquorsalesrate_container = .205;
      AltLiquorsalesrate_container = .205;
      Liquorliterrate = 2.4408;
      AltLiquorliterrate = 2.4408;
            liquorsalestax_container =
liquorsalesbase_container*Liquorsalesrate_container;
            liquorvoltax_container = liquorvolbase_container *
Liquorliterrate;
            Altliquorsalestax_container =
liquorsalesbase_container*AltLiquorsalesrate_container;
            Altliquorvoltax_container = liquorvolbase_container *
AltLiquorliterrate;

      *Liquor sales and liter by the drink;
      Liquorsalesrate_drink = .137;
      AltLiquorsalesrate_drink = .137;
            liquorsalestax_drink =
.15*liquorsalesbase_drink*Liquorsalesrate_drink;
            liquorvoltax_drink = liquorvolbase_drink * Liquorliterrate;
            Altliquorsalestax_drink =
.15*liquorsalesbase_drink*AltLiquorsalesrate_drink;
            Altliquorvoltax_drink = liquorvolbase_drink *
AltLiquorliterrate;

      *Wine tax;
      Wineliterrate = .2292;
      AltWineliterrate = .2292;
            Winelitertax = Wineliterbase*wineliterrate;
            AltWinelitertax = Wineliterbase*altwineliterrate;

      *Beer Tax - rate per 31 gallons;
      Beertaxrate = 8.08;
      AltBeertaxrate = 8.08;
            Beertax = beerbase * beertaxrate;
            AltBeertax = beerbase * altbeertaxrate;

*Insurance Tax;
      Insrate = .02;
      Altinsrate = .02;
            Instax = insrate*insbase;
            AltInstax = altinsrate*insbase;
```

```
*Cigarette and other tobacco products taxes;
      Cigrate = 1.425;
      Altcigrate = 1.425;
            cigtax = cigbase*cigrate;
            Altcigtax = cigbase*altcigrate;
      Othertobrate = 1.294;
      Altothertobrate = 1.294;
            othertobtax = othertobbase*othertobrate;
            altothertobtax = othertobbase*altothertobrate;

*Public Utility Tax;
      ElecPUTrate = .0378+.03201;
      AltElecPUTrate = .0378+.03201;
            ElecPUT = ElecPUTrate*elecbase;
            AltElecPUT = AltElecPUTrate*elecbase;
      NatgasPUTrate = .03852+.03046;
      AltNatgasPUTrate = .03852+.03046;
            NatgasPUT = NatgasPUTrate*naturalgasbase;
            AltNatgasPUT = AltNatgasPUTrate*naturalgasbase;
      WaterseweragePUTrate = (.05029+.03852)/2+((.0535+.17192)/2);
      AltWaterseweragePUTrate = (.05029+.03852)/2+((.0535+.17192)/2);
            WaterseweragePUT = WaterseweragePUTrate*waterseweragebase;
            AltWaterseweragePUT = AltWaterseweragePUTrate*waterseweragebase;
      GarbagePUTrate = .036+.06372;
      AltGarbagePUTrate = .036+.06372;
            GarbagePUT = GarbagePUTrate*garbagebase;
            AltGarbagePUT = AltGarbagePUTrate*garbagebase;
      IntercityPUTrate = .01926;
      AltIntercityPUTrate = .01926;
            IntercityPUT = IntercityPUTrate*intercitybase;
            AltIntercityPUT = AltIntercityPUTrate*intercitybase;
      IntracityPUTrate = .01926;
      AltIntracityPUTrate = .01926;
            IntracityPUT = IntracityPUTrate*intracitybase;
            AltIntracityPUT = AltIntracityPUTrate*intracitybase;

*Gas Tax;
      Gasrate = .23;
      Altgastax = .23;
            Gastax = gasrate*gasbase;
            Altgastax = altgastax*gasbase;

Alcoholtaxes = liquorsalestax_container + liquorvoltax_container +
                        liquorsalestax_drink + liquorvoltax_drink +
Winelitertax + Beertax;
AltAlcoholtaxes = Altliquorsalestax_container + Altliquorvoltax_container +
                        Altliquorsalestax_drink + Altliquorvoltax_drink +
AltWinelitertax + AltBeertax;
Tobaccotaxes = cigtax + othertobtax;
AltTobaccotaxes = Altcigtax + altothertobtax;
UtilityTaxes =  ElecPUT + NatgasPUT + WaterseweragePUT + GarbagePUT +
                        IntercityPUT + IntracityPUT;
AltUtilityTaxes = AltElecPUT + AltNatgasPUT + AltWaterseweragePUT +
                        AltGarbagePUT + AltIntercityPUT + AltIntracityPUT;

TotalExciseTaxes = salestax + Alcoholtaxes + Instax + Tobaccotaxes +
UtilityTaxes + Gastax;
AltTotalExciseTaxes = alt1tax + Altalcoholtaxes + AltInstax +
AltTobaccotaxes +
```

```
AltUtilityTaxes + Altgastax;
run;

*-------------------------------------------------------------------------;
*Merge aggregated tax with household characteristics from WAPOP;
*Run pgm called 'Second Earner Wages for Married Households'
to create data extaxmdl.secondwage;
*-------------------------------------------------------------------------;

data extaxmdl.excisetaxmerged&x;
merge excisetaxes&x z extaxmdl.secondwage;
by id;
if earned2 = . then earned2=0;
households=1;
income=hhinc;

*-------------------------------------------------------------------------;
*Income Tax Calculations;
*-------------------------------------------------------------------------;

*Earned Income Credit Calculation;
if nwageinc in (.A, .D, .N, .R, .S) and hhearn99 not in (.A, .D, .N, .R, .S)
 then nwageinc = hhinc - hhearn99;
if hhearn99 in (.A, .D, .N, .R, .S) and nwageinc not in (.A, .D, .N, .R, .S)
 then hhearn99 = hhinc - nwageinc;
if hhearn99 in (.A, .D, .N, .R, .S) and nwageinc in (.A, .D, .N, .R, .S)
 then hhearn99 = hhinc;

If nwageinc<2450 then do;
      if chldrn20 = 0 then do;
            if 25<= age <65 and 1< hhearn99 < 10711 then
            EarnIncCredit = Min(hhearn99*.0765, 4758*.0765) -
                        Max(0,(hhearn99-6000)*.0765);
            Else EarnIncCredit = 0;
            End;
      if chldrn20 > 0 then do;
            If chldrn20=1 and 1< hhearn99 < 28281 then
                  EarnIncCredit = Min(hhearn99*.34, 7141*.34) -
                        Max(0,(hhearn99-13100)*.1598);
            If chldrn20>1 and 1< hhearn99 < 32121 then
                  EarnIncCredit = Min(hhearn99*.4, 10020*.4) -
                        Max(0,(hhearn99-13100)*.2106);
            Else EarnIncCredit = 0;
            End;
      Else EarnIncCredit = 0;
      End;
Else EarnIncCredit = 0;


*Elderly calculations;

elderly=0;
if age>64 then elderly = 1;
if age>64 and hhtype=1 then elderly=2;
if q4p4g =1 then elderly =1;

*Establish income tax filer type;

If hhtype in (1) then filertype=1;
```

```
If hhtype=2 then do;
      if ranuni(567) < .72 then filertype=2;
      else filertype = 3;
      end;
if hhtype not in (1,2) then filertype=3;

*Itemized deduction calculations;
*Households are selected to be itemizers based on share of
itemizers within each income category;
*Data on itemizers and amount from DOR income tax simulation model;

If filertype=1 then do;
      if hhinc<20000 then do;
            if ranuni(123)<.13 then itemized = 1*hhinc;
            else itemized = 0;
            end;
      if 20000<=hhinc<30000 then do;
            if ranuni(123)<.209 then itemized = .554*hhinc;
            else itemized = 0;
            end;
      if 30000<=hhinc<40000 then do;
            if ranuni(123)<.284 then itemized = .41*hhinc;
            else itemized = 0;
            end;
      if 40000<=hhinc<50000 then do;
            if ranuni(123)<.411 then itemized = .296*hhinc;
            else itemized = 0;
            end;
      if 50000<=hhinc<60000 then do;
            if ranuni(123)<.549 then itemized = .247*hhinc;
            else itemized = 0;
            end;
      if 60000<=hhinc<70000 then do;
            if ranuni(123)<.612 then itemized = .221*hhinc;
            else itemized = 0;
            end;
      if 70000<=hhinc<80000 then do;
            if ranuni(123)<.688 then itemized = .202*hhinc;
            else itemized = 0;
            end;
      if 80000<=hhinc<100000 then do;
            if ranuni(123)<.761 then itemized = .19*hhinc;
            else itemized = 0;
            end;
      if 100000<=hhinc<130000 then do;
            if ranuni(123)<.824 then itemized = .177*hhinc;
            else itemized = 0;
            end;
      if hhinc>=130000 then do;
            if ranuni(123)<.801 then itemized = .072*hhinc;
            else itemized = 0;
            end;
end;

If filertype=2 then do;
      if hhinc<20000 then do;
            if ranuni(123)<.025 then itemized = .735*hhinc;
            else itemized = 0;
            end;
```

```
      if 20000<=hhinc<30000 then do;
            if ranuni(123)<.106 then itemized = .501*hhinc;
            else itemized = 0;
            end;
      if 30000<=hhinc<40000 then do;
            if ranuni(123)<.322 then itemized = .334*hhinc;
            else itemized = 0;
            end;
      if 40000<=hhinc<50000 then do;
            if ranuni(123)<.432 then itemized = .252*hhinc;
            else itemized = 0;
            end;
      if 50000<=hhinc<60000 then do;
            if ranuni(123)<.477 then itemized = .232*hhinc;
            else itemized = 0;
            end;
      if 60000<=hhinc<70000 then do;
            if ranuni(123)<.658 then itemized = .204*hhinc;
            else itemized = 0;
            end;
      if 70000<=hhinc<80000 then do;
            if ranuni(123)<.70 then itemized = .17*hhinc;
            else itemized = 0;
            end;
      if 80000<=hhinc<100000 then do;
            if ranuni(123)<.765 then itemized = .151*hhinc;
            else itemized = 0;
            end;
      if 100000<=hhinc<130000 then do;
            if ranuni(123)<.836 then itemized = .161*hhinc;
            else itemized = 0;
            end;
      if hhinc>=130000 then do;
            if ranuni(123)<.737 then itemized = .102*hhinc;
            else itemized = 0;
            end;
end;

If filertype=3 then do;
      if hhinc<20000 then do;
            if ranuni(123)<.049 then itemized = 1*hhinc;
            else itemized = 0;
            end;
      if 20000<=hhinc<30000 then do;
            if ranuni(123)<.141 then itemized = .40*hhinc;
            else itemized = 0;
            end;
      if 30000<=hhinc<40000 then do;
            if ranuni(123)<.239 then itemized = .297*hhinc;
            else itemized = 0;
            end;
      if 40000<=hhinc<50000 then do;
            if ranuni(123)<.386 then itemized = .271*hhinc;
            else itemized = 0;
            end;
      if 50000<=hhinc<60000 then do;
            if ranuni(123)<.506 then itemized = .219*hhinc;
            else itemized = 0;
            end;
```

```
      if 60000<=hhinc<70000 then do;
            if ranuni(123)<.568 then itemized = .203*hhinc;
            else itemized = 0;
            end;
      if 70000<=hhinc<80000 then do;
            if ranuni(123)<.592 then itemized = .198*hhinc;
            else itemized = 0;
            end;
      if 80000<=hhinc<100000 then do;
            if ranuni(123)<.638 then itemized = .19*hhinc;
            else itemized = 0;
            end;
      if 100000<=hhinc<130000 then do;
            if ranuni(123)<.666 then itemized = .164*hhinc;
            else itemized = 0;
            end;
      if hhinc>=130000 then do;
            if ranuni(123)<.664 then itemized = .085*hhinc;
            else itemized = 0;
            end;
end;

*Alternative Definitions of taxed income;

*Taxedincome = max(0,hhinc);
*Taxedincome = max(0,(hhinc - 5000*peopl));
*Taxedincome = max(0,(hhinc - 20000));

*Calculation allowing variable std ded for two earners;

If filertype in (1) then
      Taxedincome = max(0,hhinc-(7000+min(earned2,3000))
                        -1000*elderly-2900*peopl);
If filertype in (2) then
      Taxedincome = max(0,hhinc-(7000)
                        -1000*elderly-2900*peopl);
If filertype in (3) then
      Taxedincome = max(0,hhinc-(5000)
                        -1000*elderly-2900*peopl);

*Calculation allowing fed std ded or itemization plus
fed personal exemption estimated for 2005;

*If filertype in (1) then
      Taxedincome = max(0,hhinc-max(8350,itemized)
                        -3175*peopl);
*If filertype in (2) then
      Taxedincome = max(0,hhinc-max(7350,itemized)
                        -3175*peopl);
*If filertype in (3) then
      Taxedincome = max(0,hhinc-max(4975,itemized)
                        -3175*peopl);

*Flat Rate Income Tax Calculation;

FlatrateIncTax=1*Taxedincome;

*FlatrateIncTax=max(0,.05*Taxedincome-EarnIncCredit*.1);
```

```
*Graduated Rate Income Tax;

Rate1=.02;
Rate2=.03;
Rate3=.05;
If filertype in (1) then do;
      Bracket1 = min(Taxedincome,49900);
      Bracket2 = min(120650-49900,max(0,Taxedincome-49900));
      Bracket3 = max(0,Taxedincome-120650);
End;
If filertype in (2) then do;
      Bracket1 = min(Taxedincome,37425);
      Bracket2 = min(90487.5-37425,max(0,Taxedincome-37425));
      Bracket3 = max(0,Taxedincome-90487.5);
End;
If filertype not in (1,2) then do;
      Bracket1 = min(Taxedincome,24950);
      Bracket2 = min(60325-24950,max(0,Taxedincome-24950));
      Bracket3 = max(0,Taxedincome-60325);
End;

GradRateIncTax = Rate1*Bracket1+Rate2*Bracket2+Rate3*Bracket3;
*GradRateIncTax = Max(0,Rate1*Bracket1+Rate2*Bracket2
                       +Rate3*Bracket3-EarnIncCredit*.1);

drop _freq_ _type_;
run;
%end;
%mend loop1;
%loop1;

OPTIONS ls=123 ps=52 pageno=1;

*-------------------------------------------------------------------------------;
*Calculate total tax by tax type;
*-------------------------------------------------------------------------------;

%macro loop2;
%do z=1 %to 7;
proc summary  data=extaxmdl.excisetaxmerged&z;
weight fnlwgt;
var salestax alt1tax alt2tax liquorsalestax_container liquorvoltax_container
Altliquorsalestax_container Altliquorvoltax_container liquorsalestax_drink
liquorvoltax_drink Altliquorsalestax_drink Altliquorvoltax_drink
Winelitertax AltWinelitertax Beertax AltBeertax Instax AltInstax
cigtax Altcigtax othertobtax altothertobtax
ElecPUT AltElecPUT NatgasPUT AltNatgasPUT WaterseweragePUT
AltWaterseweragePUT
GarbagePUT AltGarbagePUT IntercityPUT AltIntercityPUT IntracityPUT
AltIntracityPUT
Gastax Altgastax households EarnIncCredit GradRateIncTax Bracket1 Bracket2
Bracket3
FlatrateIncTax peopl;
output out=weightedtax&z sum=;
run;
%end;
%mend loop2;
%loop2;
```

```
*Average the data from the seven imputation groups;

Data one;
set weightedtax1 weightedtax2 weightedtax3 weightedtax4
weightedtax5 weightedtax6 weightedtax7;
proc summary  data=one;
var salestax alt1tax alt2tax liquorsalestax_container liquorvoltax_container
Altliquorsalestax_container Altliquorvoltax_container liquorsalestax_drink
liquorvoltax_drink Altliquorsalestax_drink Altliquorvoltax_drink
Winelitertax AltWinelitertax Beertax AltBeertax Instax AltInstax
cigtax Altcigtax othertobtax altothertobtax
ElecPUT AltElecPUT NatgasPUT AltNatgasPUT WaterseweragePUT
AltWaterseweragePUT
GarbagePUT AltGarbagePUT IntercityPUT AltIntercityPUT IntracityPUT
AltIntracityPUT
Gastax Altgastax households EarnIncCredit GradRateIncTax Bracket1 Bracket2
Bracket3
 FlatrateIncTax peopl;
output out=Aveweightedtax mean=;
run;
Title1 "Total tax amounts for all groups (Average)";
proc print data=Aveweightedtax;
var salestax alt1tax alt2tax liquorsalestax_container liquorvoltax_container
Altliquorsalestax_container Altliquorvoltax_container liquorsalestax_drink
liquorvoltax_drink Altliquorsalestax_drink Altliquorvoltax_drink
Winelitertax AltWinelitertax Beertax AltBeertax Instax AltInstax
cigtax Altcigtax othertobtax altothertobtax
ElecPUT AltElecPUT NatgasPUT AltNatgasPUT WaterseweragePUT
AltWaterseweragePUT
GarbagePUT AltGarbagePUT IntercityPUT AltIntercityPUT IntracityPUT
AltIntracityPUT
Gastax Altgastax EarnIncCredit GradRateIncTax Bracket1 Bracket2 Bracket3
 FlatrateIncTax peopl;
format salestax alt1tax alt2tax liquorsalestax_container
liquorvoltax_container
Altliquorsalestax_container Altliquorvoltax_container liquorsalestax_drink
liquorvoltax_drink Altliquorsalestax_drink Altliquorvoltax_drink
Winelitertax AltWinelitertax Beertax AltBeertax Instax AltInstax
cigtax Altcigtax othertobtax altothertobtax
ElecPUT AltElecPUT NatgasPUT AltNatgasPUT WaterseweragePUT
AltWaterseweragePUT
GarbagePUT AltGarbagePUT IntercityPUT AltIntercityPUT IntracityPUT
AltIntracityPUT
Gastax Altgastax EarnIncCredit GradRateIncTax Bracket1 Bracket2 Bracket3
 FlatrateIncTax peopl comma15.;
run;

*-------------------------------------------------------------------------;
*Calculate Tax by income group for each imputation group;
*-------------------------------------------------------------------------;

OPTIONS ls=130 ps=50 pageno=1;
%macro loop3;
%do z=1 %to 7;
proc summary nway data=extaxmdl.excisetaxmerged&z;
class hhinc ;
format hhinc incfmtd.;
weight fnlwgt;
```

```
var TotalExciseTaxes AltTotalExciseTaxes Alcoholtaxes AltAlcoholtaxes
Tobaccotaxes
AltTobaccotaxes UtilityTaxes AltUtilityTaxes
salestax alt1tax alt2tax liquorsalestax_container liquorvoltax_container
Altliquorsalestax_container Altliquorvoltax_container liquorsalestax_drink
liquorvoltax_drink Altliquorsalestax_drink Altliquorvoltax_drink
Winelitertax AltWinelitertax Beertax AltBeertax Instax AltInstax
cigtax Altcigtax othertobtax altothertobtax
ElecPUT AltElecPUT NatgasPUT AltNatgasPUT WaterseweragePUT
AltWaterseweragePUT
GarbagePUT AltGarbagePUT IntercityPUT AltIntercityPUT IntracityPUT
AltIntracityPUT
Gastax Altgastax households EarnIncCredit GradRateIncTax Bracket1 Bracket2
Bracket3
 FlatrateIncTax peopl;
output out=weightedtaxbyinc&z sum=;
run;
%end;
%mend loop3;
%loop3;

*Add tax from each group and take average;

Data two;
set weightedtaxbyinc1 weightedtaxbyinc2 weightedtaxbyinc3
weightedtaxbyinc4 weightedtaxbyinc5 weightedtaxbyinc6
weightedtaxbyinc7;
run;
Proc summary data = two nway;
class hhinc;
format hhinc incfmtd.;
var TotalExciseTaxes AltTotalExciseTaxes Alcoholtaxes AltAlcoholtaxes
Tobaccotaxes
AltTobaccotaxes UtilityTaxes AltUtilityTaxes
salestax alt1tax alt2tax liquorsalestax_container liquorvoltax_container
Altliquorsalestax_container Altliquorvoltax_container liquorsalestax_drink
liquorvoltax_drink Altliquorsalestax_drink Altliquorvoltax_drink
Winelitertax AltWinelitertax Beertax AltBeertax Instax AltInstax
cigtax Altcigtax othertobtax altothertobtax
ElecPUT AltElecPUT NatgasPUT AltNatgasPUT WaterseweragePUT
AltWaterseweragePUT
GarbagePUT AltGarbagePUT IntercityPUT AltIntercityPUT IntracityPUT
AltIntracityPUT
Gastax Altgastax households EarnIncCredit GradRateIncTax Bracket1 Bracket2
Bracket3
 FlatrateIncTax peopl;
output out=Aveweightedtaxbyinc mean=;
run;

*Divide by number of households by income group
and calc avg tax per household;

data avgtax;
set Aveweightedtaxbyinc;
AvgTotalExciseTaxes = TotalExciseTaxes/households;
AvgAltTotalExciseTaxes = AltTotalExciseTaxes/households;
Avgsalestax = salestax/households;
Avgalt1tax = alt1tax/households;
Avgalt2tax = alt2tax/households;
```

```
avgAlcoholtaxes = Alcoholtaxes/households;
avgAltAlcoholtaxes = AltAlcoholtaxes/households;
avgInstax = instax/households;
avgAltinstax = altinstax/households;
avgTobaccotaxes = Tobaccotaxes/households;
avgAltTobaccotaxes = AltTobaccotaxes/households;
avgUtilityTaxes = UtilityTaxes/households;
avgAltUtilityTaxes = AltUtilityTaxes/households;
avgGastax = gastax/households;
avgAltgastax = altgastax/households;
avgEarnIncCredit= EarnIncCredit/households;
avgGradRateIncTax= GradRateIncTax/households;
avgBracket1= Bracket1/households;
avgBracket2= Bracket2/households;
avgBracket3= Bracket3/households;
avgFlatrateIncTax= FlatrateIncTax/households;
avgpeopl = peopl/households;
drop _freq_ _type_;
run;

title1 "Average Tax for all Groups (Average)" ;

Proc tabulate data = avgtax;
class hhinc;
format hhinc incfmtd.;
var AvgTotalExciseTaxes AvgAltTotalExciseTaxes Avgsalestax Avgalt1tax
Avgalt2tax
avgAlcoholtaxes avgAltAlcoholtaxes avgInstax avgAltinstax
avgTobaccotaxes avgAltTobaccotaxes
avgUtilityTaxes avgAltUtilityTaxes avgGastax avgAltgastax
avgEarnIncCredit avgGradRateIncTax avgBracket1 avgBracket2 avgBracket3
 avgFlatrateIncTax avgpeopl;
table hhinc,(AvgTotalExciseTaxes AvgAltTotalExciseTaxes Avgsalestax
Avgalt1tax Avgalt2tax
avgAlcoholtaxes avgAltAlcoholtaxes avgInstax avgAltinstax
avgTobaccotaxes avgAltTobaccotaxes
avgUtilityTaxes avgAltUtilityTaxes avgGastax avgAltgastax
avgEarnIncCredit avgGradRateIncTax avgBracket1 avgBracket2 avgBracket3
 avgFlatrateIncTax)*f=comma10. avgpeopl*f=comma10.2;
run;

*Merge in total income by income group
and calc avg tax as percent of income;

data avgtax;
merge aveweightedtaxbyinc incomebyclass;
by hhinc;
AvgTotalExciseTaxes = TotalExciseTaxes/income;
AvgAltTotalExciseTaxes = AltTotalExciseTaxes/income;
Avgsalestax = salestax/income;
Avgalt1tax = alt1tax/income;
Avgalt2tax = alt2tax/income;
avgAlcoholtaxes = Alcoholtaxes/income;
avgAltAlcoholtaxes = AltAlcoholtaxes/income;
avgInstax = instax/income;
avgAltinstax = altinstax/income;
avgTobaccotaxes = Tobaccotaxes/income;
avgAltTobaccotaxes = AltTobaccotaxes/income;
avgUtilityTaxes = UtilityTaxes/income;
```

```
avgAltUtilityTaxes = AltUtilityTaxes/income;
avgGastax = gastax/income;
avgAltgastax = altgastax/income;
avgEarnIncCredit= EarnIncCredit/income;
avgGradRateIncTax= GradRateIncTax/income;
avgBracket1= Bracket1/income;
avgBracket2= Bracket2/income;
avgBracket3= Bracket3/income;
avgFlatrateIncTax= FlatrateIncTax/income;
drop _freq_ _type_;
run;


title1 "Tax as Percent of Income for all Groups (Average)" ;

Proc tabulate data = avgtax ;
class hhinc;
format hhinc incfmtd.;
var AvgTotalExciseTaxes AvgAltTotalExciseTaxes Avgsalestax Avgalt1tax
Avgalt2tax
avgAlcoholtaxes avgAltAlcoholtaxes avgInstax avgAltinstax
avgTobaccotaxes avgAltTobaccotaxes
avgUtilityTaxes avgAltUtilityTaxes avgGastax avgAltgastax
avgEarnIncCredit avgGradRateIncTax avgBracket1 avgBracket2 avgBracket3
 avgFlatrateIncTax;
table hhinc,(AvgTotalExciseTaxes AvgAltTotalExciseTaxes Avgsalestax
Avgalt1tax Avgalt2tax
avgAlcoholtaxes avgAltAlcoholtaxes avgInstax avgAltinstax
avgTobaccotaxes avgAltTobaccotaxes
avgUtilityTaxes avgAltUtilityTaxes avgGastax avgAltgastax
avgEarnIncCredit avgGradRateIncTax avgBracket1 avgBracket2 avgBracket3
 avgFlatrateIncTax)*f=percent10.2;
run;


*************************************************************************;
* program:    Second Earner Wages for Married Households                 ;
* programmer: Rick Peterson                                              ;
* project:    Washington Excise Tax Microsimulation Model                ;
* date:       Nov 21, 2002                                               ;
*                                                                        ;
* purpose:    Calculates relationship between first and second earner     ;
*             wages for married households.  Creates frequency           ;
*             distribution for use in the Department of Revenue's        ;
*             Income Tax Model and data set for use in 'Tax Calculator   ;
*             for SimTax Model.                                          ;
;
*------------------------------------------------------------------------;
* libraries:   extaxmdl - location of excise tax model data sets         ;
*              popsur - location of Washington Population Survey data    ;
*------------------------------------------------------------------------;
* incoming:    popsur.sps00f04 Washington Population Survey              ;
*------------------------------------------------------------------------;
* formats:     'Formats for microsimulation model 1'                     ;
*              'Formats for microsimulation model 2'                     ;
*------------------------------------------------------------------------;
* outgoing:    extaxmdl.secondwage - for use in 'Tax Calculator for      ;
*              SimTax Model'                                             ;
*------------------------------------------------------------------------;
```

```
* reports:     Frequency Distributin of Ratio of Second Wage Earner's     ;
*              Wage to 1st Earner's Wage by income class                  ;
*-------------------------------------------------------------------------;
* changes:                                                                 ;
*                                                                         ;
*-------------------------------------------------------------------------;
* notes:
***************************************************************************;

*-------------------------------------------------------------------------;
*Read married household data from WAPOP;
*-------------------------------------------------------------------------;

Data x;
set popsur.sps00f04;
Where hhtype =1;
earned = 0;
if q6p1a not in (.A, .D, .N, .R, .S) and q6p4a not in (.A, .D, .N, .R, .S)
 then earned = q6p1a + q6p4a;
if q6p1a in (.A, .D, .N, .R, .S) and q6p4a not in (.A, .D, .N, .R, .S)
 then earned =  q6p4a;
if q6p1a not in (.A, .D, .N, .R, .S) and q6p4a in (.A, .D, .N, .R, .S)
 then earned = q6p1a;

keep id pnum fnlwgt hhearn99
q2p10 hhtype earned q6p1a q6p4a hhinc nwageinc;
run;

Proc sort data=x out=xx;
by id decending earned ;
run;

data xxx a;
set xx ;
by id;
if first.id eq 1 then output a;
if first.id ne 1 then output xxx;
run;

Proc summary data =xxx;
by id;
var earned ;
output out=xxxx sum=earned2;
run;

data aa aaa;
merge a xxxx;
by id;
if earned > 0 then do;
      ratio = earned2/earned;
      output aa;
      end;
else do;
      ratio = 0;
      output aaa;
      end;
label ratio='Ratio of 2nd earner wage to 1st earner wage';
run;
Title1 "Ratio of Second Wage Earner's Wage to 1st Earner's Wage";
```

```
footnote;
Proc format;
 value ratiofmt
                    low -<.01 = 'Zero'
                     .01-<.25 ='Zero+ to 25%'
               .25-<.5   ='25% to 50%'
               .5-<.75   ='50% to 75%'
               .75-high  ='Over 75%';
                    run;

Title1 h=2 "Ratio of Second Wage Earner's Wage to 1st Earner's Wage";
footnote;

Proc freq data=aa;
table hhinc*ratio / nocol nocum nofreq out=earn2ratio;
weight fnlwgt;
format hhinc incfmtd.;
format ratio ratiofmt.;
run;

*-------------------------------------------------------------------------------;
*Create data set to assign earned income to 1st
and 2nd worker in married households without wage data;
*-------------------------------------------------------------------------------;

Proc summary data=earn2ratio;
by hhinc;
format hhinc incfmtd.;
var count;
output out=sumearn sum=total;
run;


data bb;
Merge earn2ratio sumearn;
by hhinc;
format hhinc incfmtd.;
ratio2=count/total;
keep hhinc ratio2 ratio;
run;

*Transpose the data making each percentage a variable;
*Each percentage represents the share of households with ratios
 equal to the ranges assigned to ratiofmt above;

Proc transpose  name=prob prefix=p data=bb out=bbb;
var ratio2;
run;

*Merge the prob data with the married households without
data on wage earnings;
*Assign earnings for 1st earner and 2nd earner based on
above data;

data B;
if _n_=1 then set bbb;
set aaa;
array r{5} r1-r5 (0 .125 .375 .625 .875);
retain p1-p50 r1-r5;
```

```
if nwageinc not in (.A, .D, .N, .R, .S)
then hhearn99 = hhinc - nwageinc;
else hhearn99 = hhinc;

if hhinc<20000 then do;
y = rantbl(25,of p1-p5);
Ratio= r(y);
earned=hhearn99/(1+ratio);
earned2 = ratio*earned;
end;

if 20000<=hhinc<30000 then do;
y = rantbl(25,of p6-p10);
Ratio= r(y);
earned=hhearn99/(1+ratio);
earned2 = ratio*earned;
end;

if 30000<=hhinc<40000 then do;
y = rantbl(25,of p11-p15);
Ratio= r(y);
earned=hhearn99/(1+ratio);
earned2 = ratio*earned;
end;

if 40000<=hhinc<50000 then do;
y = rantbl(25,of p16-p20);
Ratio= r(y);
earned=hhearn99/(1+ratio);
earned2 = ratio*earned;
end;

if 50000<=hhinc<60000 then do;
y = rantbl(25,of p21-p25);
Ratio= r(y);
earned=hhearn99/(1+ratio);
earned2 = ratio*earned;
end;

if 60000<=hhinc<70000 then do;
y = rantbl(25,of p26-p30);
Ratio= r(y);
earned=hhearn99/(1+ratio);
earned2 = ratio*earned;
end;

if 70000<=hhinc<80000 then do;
y = rantbl(25,of p31-p35);
Ratio= r(y);
earned=hhearn99/(1+ratio);
earned2 = ratio*earned;
end;

if 80000<=hhinc<100000 then do;
y = rantbl(25,of p36-p40);
Ratio= r(y);
earned=hhearn99/(1+ratio);
earned2 = ratio*earned;
```

```
end;

if 100000<=hhinc<130000 then do;
y = rantbl(25,of p41-p45);
Ratio= r(y);
earned=hhearn99/(1+ratio);
earned2 = ratio*earned;
end;

if hhinc>=130000 then do;
y = rantbl(25,of p46-p50);
Ratio= r(y);
earned=hhearn99/(1+ratio);
earned2 = ratio*earned;
end;

if nwageinc not in (.A, .D, .N, .R, .S) then do;
      test = hhinc-nwageinc-earned-earned2;
      test2 = hhearn99 - earned - earned2;
      end;
else do;
      test = hhinc-earned-earned2;
      test2 = hhearn99 - earned - earned2;
      end;
keep id hhinc hhearn99 nwageinc earned earned2
test test2;
run;

*Create data set for married families with earnings for
2nd earner for use in Tax Calculator for SimTax Model;

data extaxmdl.secondwage;
set aa b;
keep id earned2;
run;

Proc sort data=extaxmdl.secondwage;
by id;
run;
```

Ratio of Second Wage Earner's Wage to 1st Earner's Wage

The FREQ Procedure

Percent Row Pct

Table of HHINC by ratio

HHINC(1999 HOUSEHOLD TOTAL INCOME)	ratio(Ratio of 2nd earner wage to 1st earner wage)					Total
	Zero	Zero+ to 25%	25% to 50%	50% to 75%	Over 75%	
$0 to $20,000	3.03 81.19	0.08 2.20	0.33 8.86	0.24 6.35	0.05 1.39	3.73
$20,000 to $30,000	3.50 65.07	0.96 17.82	0.31 5.83	0.22 4.15	0.38 7.13	5.38
$30,000 to $40,000	4.79 58.53	1.12 13.69	0.88 10.80	0.38 4.62	1.01 12.36	8.18
$40,000 to $50,000	4.52 40.25	2.21 19.67	1.42 12.68	1.55 13.81	1.52 13.58	11.22
$50,000 to $60,000	4.59 38.34	1.33 11.14	2.25 18.81	1.41 11.79	2.38 19.92	11.96
$60,000 to $70,000	3.36 29.81	1.57 13.93	1.40 12.40	2.34 20.70	2.61 23.17	11.29
$70,000 to $80,000	2.83 28.90	1.78 18.12	1.53 15.62	0.98 9.98	2.69 27.38	9.81
$80,000 to $100,000	3.11 21.18	1.78 12.14	2.17 14.80	3.01 20.47	4.62 31.42	14.69
$100,000 to $130,000	2.50 22.15	1.22 10.84	1.51 13.40	2.56 22.72	3.49 30.89	11.29
Over $130,000	4.00 32.15	2.20 17.70	2.58 20.70	1.52 12.24	2.14 17.21	12.45
Total	298572 36.24	117500 14.26	118603 14.39	117073 14.21	172179 20.90	823927 100.00