

BUAN6337 Predictive Analytics Using SAS - F20

Homework Assignment 1

Question 1

The World Health Organization (WHO) collected data in countries across the world regarding the outbreak of swine flu cases and deaths in 2009. The data in the file SwineFlu2009.dat include counts per country by month during the epidemic.

a. First, examine the raw data file SwineFlu2009.dat using Notepad.

when open with Notepad, the dataset is without variables name, and I can easily find out that there are some data are missing.

b. Write a DATA step to read the file into SAS. Make sure that each variable is assigned a unique (and descriptive) name and is of the correct type – character or numeric.

```
1 data Buan6337.SwineFlu2009;
2 infile 'SwineFlu2009.dat';
3 input FirstCase_ID 1-4
4       ContinentCase_ID 13-17
5       Country $27-60
6       First_Case_Date yymmdd10.
7       Apr_Cases 79-88
8       May_Cases 89-98
9       Jun_Cases 99-108
10      Jul_Cases 108-118
11      Aug_Cases 119-128
12      Case_Cum 129-144
13      FirstDeath_ID 145-154
14      ContinentDeath_ID 155-168
15      First_Death_Date yymmdd10.
16      May_Deaths 194-203
17      Jun_Deaths 204-213
18      Jul_Deaths 214-223
19      Aug_Deaths 224-233
20      Sep_Deaths 234-243
21      Oct_Deaths 244-255
22      Nov_Deaths 256-264
23      Dec_Deaths;
```

- c. After the dataset is created, locate the file in the work library through the explorer window in SAS. Double click on the dataset to view the data.

	FirstCase_ID	ContinentCase_ID	Country	First_Case_Date	Apr_Cases	May_Cases	Jun_Cases	Jul_Cases	Aug_Cases	Case_Cum	FirstDeath_ID	ContinentDeath_ID	First_Death
1	137	3.35	Afghanistan	2009-07-08	-	-	-	-	32	32	99	5.33	*****
2	154	2.43	Albania	2009-07-22	-	-	-	-	3	13	-	-	-
3	161	2.44	Andorra	2009-06-29	-	-	-	-	1	1	-	-	-
4	101	1.22	Antigua and Barbuda	2009-06-24	-	-	-	2	3	4	-	-	-
5	122	1.24	Aruba	2009-06-03	-	-	-	-	13	13	-	-	-
6	9	2.03	Austria	2009-04-29	-	1	1	15	153	192	100	3.24	*****
7	163	2.46	Azerbaijan	2009-07-31	-	-	-	-	2	2	108	3.3	*****
8	59	1.11	Bahamas	2009-06-01	-	-	1	6	29	29	86	1.16	*****
9	48	3.12	Bahrain	2009-05-27	-	-	1	15	123	148	68	5.27	*****
10	96	3.26	Bangladesh	2009-06-22	-	-	-	1	30	32	72	5.28	*****
11	68	4.14	Barbados	2009-06-05	-	-	-	10	23	39	81	1.14	*****
12	-	-	Belarus	-	-	-	-	-	-	-	102	3.26	*****
13	34	2.16	Belgium	2009-05-15	-	-	12	707	1586	1586	40	3.04	*****
14	147	1.3	Belize	2009-07-16	-	-	-	-	23	27	-	-	-
15	156	3.36	Bhutan	2009-06-27	-	-	-	-	2	3	-	-	-
16	124	2.39	Bosnia and Herzegovina	2009-06-03	-	-	-	-	2	10	114	3.33	*****
17	145	6.16	Botswana	2009-07-13	-	-	-	-	13	23	-	-	-
18	81	1.19	British Virgin Islands	2009-06-17	-	-	-	1	5	8	-	-	-
19	65	2.26	Bulgaria	2009-06-03	-	-	-	-	36	40	82	3.15	*****
20	107	3.28	Cambodia	2009-06-24	-	-	-	6	17	24	80	5.29	*****
21	3	1.03	Canada	2009-04-27	-	34	1336	7983	10449	10449	3	1.03	*****
22	100	6.05	Cape Verde	2009-06-24	-	-	-	3	6	6	-	-	-
23	40	5.06	Chile	2009-05-18	-	-	250	6211	11860	12030	5	2.01	*****
24	12	3.02	China	2009-05-01	-	1	52	1518	6628	7931	26	5.04	*****
25	20	5.01	Colombia	2009-05-04	-	-	20	93	270	275	7	2.02	*****
26	135	4.09	Cook Islands	2009-07-06	-	-	-	-	1	18	59	4.05	*****
27	16	1.04	Costa Rica	2009-05-02	-	-	37	279	668	798	4	1.04	*****
28	102	6.06	Cote d'Ivoire	2009-06-24	-	-	-	2	2	2	-	-	-
29	33	1.08	Cuba	2009-05-13	-	-	4	46	234	264	84	1.15	*****
30	60	2.23	Cyprus	2009-06-01	-	-	-	1	48	297	-	-	-
31	53	2.21	Czech Republic	2009-05-29	-	-	1	9	116	162	97	3.22	*****
32	13	2.07	Denmark	2009-05-01	-	1	1	55	254	354	118	3.36	*****
33	73	1.16	Dominica	2009-06-08	-	-	-	1	1	1	-	-	-
34	35	5.04	Ecuador	2009-05-16	-	-	39	163	696	881	24	2.08	*****
35	66	6.01	Egypt	2009-06-03	-	-	-	39	283	314	29	6.01	*****
36	19	1.05	El Salvador	2009-05-04	-	-	77	736	656	656	21	1.08	*****

- d. In SAS, dates can be stored as a special type of numeric data. Modify the DATA step to make sure that the dates are read in the correct SAS date format (not as character).

SAS will detect the date, and I informat and format the data by following code.

```
/*
input
    First_Case_Date yymmdd10.
    First_Death_Date yymmdd10.;

format First_Case_Date yymmdd10.;
format First_Death_Date yymmdd10.;

*/
```

- e. Create a permanent label for each variable based on the preceding descriptions.

```
DATA Buan6337.SwineFlu2009;
SET Buan6337.SwineFlu2009;
LABEL FirstCase_ID ="ID for sorting by first case date"
    ContinentCase_ID ="X represents continent and YY represents the YYth country with the next 1st"
    Country ="Country"
    First_Case_Date ="Date of first case reported"
    Apr_Cases ="Number of cumulative cases reported on the first day of April"
    May_Cases ="Number of cumulative cases reported on the first day of May"
    Jun_Cases ="Number of cumulative cases reported on the first day of June"
    Jul_Cases ="Number of cumulative cases reported on the first day of July"
    Aug_Cases ="Number of cumulative cases reported on the first day of August"
    Case_Cum ="Last reported cumulative number of cases reported to WHO as of August 9, 2009"
    FirstDeath_ID ="ID for sorting by first death date"
    ContinentDeath_ID ="X represents continent and YY represents the YYth country with the next 1:"
    First_Death_Date ="Date of first death"
    May_Deaths ="Number of cumulative deaths reported on the first day of May"
    Jun_Deaths ="Number of cumulative deaths reported on the first day of June"
    Jul_Deaths ="Number of cumulative deaths reported on the first day of July"
    Aug_Deaths ="Number of cumulative deaths reported on the first day of August"
    Sep_Deaths ="Number of cumulative deaths reported on the first day of September"
    Oct_Deaths ="Number of cumulative deaths reported on the first day of October"
    Nov_Deaths ="Number of cumulative deaths reported on the first day of November"
    Dec_Deaths ="Number of cumulative deaths reported on the first day of December";
RUN;
```

- f. Print a report that describes the contents of the data set including the labels that you've created and other attributes of the variables.

Alphabetic List of Variables and Attributes					
#	Variable	Type	Len	Format	Label
5	Apr_Cases	Num	8		Number of cumulative cases reported on the first day of April
9	Aug_Cases	Num	8		Number of cumulative cases reported on the first day of August
17	Aug_Deaths	Num	8		Number of cumulative deaths reported on the first day of August
10	Case_Cum	Num	8		Last reported cumulative number of cases reported to WHO as of August 9, 2009
2	ContinentCase_ID	Num	8		X represents continent and YY represents the YYth country with the next 1st case
12	ContinentDeath_ID	Num	8		X represents continent and YY represents the YYth country with the next 1st death
3	Country	Char	34		Country
21	Dec_Deaths	Num	8		Number of cumulative deaths reported on the first day of December
1	FirstCase_ID	Num	8		ID for sorting by first case date
11	FirstDeath_ID	Num	8		ID for sorting by first death date
4	First_Case_Date	Num	8	YYMMDD10.	Date of first case reported
13	First_Death_Date	Num	8	YYMMDD10.	Date of first death
8	Jul_Cases	Num	8		Number of cumulative cases reported on the first day of July
16	Jul_Deaths	Num	8		Number of cumulative deaths reported on the first day of July
7	Jun_Cases	Num	8		Number of cumulative cases reported on the first day of June
15	Jun_Deaths	Num	8		Number of cumulative deaths reported on the first day of June
6	May_Cases	Num	8		Number of cumulative cases reported on the first day of May
14	May_Deaths	Num	8		Number of cumulative deaths reported on the first day of May
20	Nov_Deaths	Num	8		Number of cumulative deaths reported on the first day of November
19	Oct_Deaths	Num	8		Number of cumulative deaths reported on the first day of October
18	Sep_Deaths	Num	8		Number of cumulative deaths reported on the first day of September

Question 2

A gourmet pizza restaurant is considering adding new toppings to its menu. Each month they survey 10 customers about their preferences for three different toppings. They want data on several different toppings, so they don't always ask about the same three toppings. Customers rate each topping on a scale of 1 (would never order) to 5 (would order often). The restaurant wants to compute average ratings for all toppings, so the ratings variables need to be numeric. The raw data file Pizza.csv has variables for the respondent's ID, and the ratings for five different toppings: arugula, pine nuts, roasted butternut squash, shrimp, and grilled eggplant. The first two digits in the ID correspond to the month of the survey.

a. Examine the raw data file Pizza.csv and read it into SAS using the IMPORT procedure.

```
proc import datafile="Pizza.csv" out=Buan6337.pizza dbms=csv replace;
    getnames=yes;
run;
```

b. Print the data set (on the results screen). Print a report that describes the contents of the data set to make sure all the variables are the correct type.

```
proc print data=Buan6337.pizza;
run;
```

```
proc contents data=Buan6337.pizza;
run;
```

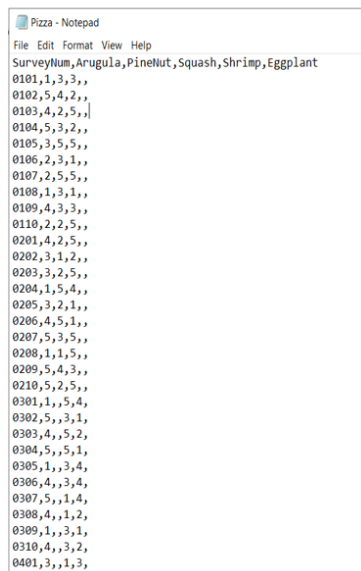
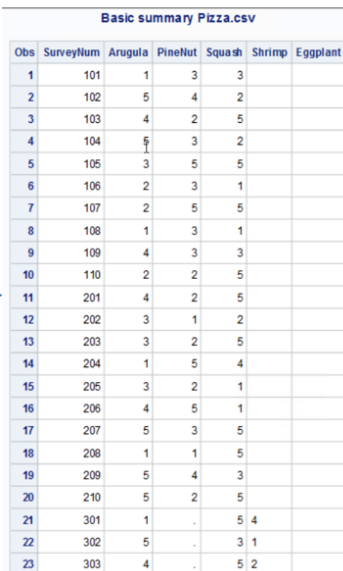
The report looks good, but the variables are not in the correct type.

The screenshot shows the SAS Results Viewer interface. On the left, a pane titled 'Results' shows a tree view with 'Print: Basic summary Pizza.csv'. The main window displays a table titled 'Basic summary Pizza.csv' with columns: Obs, SurveyNum, Arugula, PineNut, Squash, Shrimp, and Eggplant. The table contains 24 rows of data. Below the table, an 'Alphabetic List of Variables and Attributes' table is shown, listing variables and their attributes.

#	Variable	Type	Len	Format	Informat
2	Arugula	Num	8	BEST12.	BEST32.
6	Eggplant	Char	1	\$1.	\$1.
3	PineNut	Num	8	BEST12.	BEST32.
5	Shrimp	Char	1	\$1.	\$1.
4	Squash	Num	8	BEST12.	BEST32.
1	SurveyNum	Num	8	BEST12.	BEST32.

- c. Open the raw data file in a simple editor like WordPad and compare the data values to the output from part b) to make sure that they were read correctly into SAS. In a comment in your report, identify any problems with the SAS data set that cannot be resolved using the IMPORT procedure. Explain what is causing the problem.

The output looks right, but the type of two variables is not *Num*, and it means we can't do operation between the type of *Char* and *Num*.

Raw Data		SAS	
			

Read it into SAS

After checking the log, it seems variables of “Shrimp” and “Eggplant” are informat/format with Char ('\$'). And look like “proc import” can't deal with the problem that converting Characters into Numbers.

```

510  /*****
510! *****
511  *   PRODUCT:   SAS
512  *   VERSION:   9.4
513  *   CREATOR:   External File Interface
514  *   DATE:      03SEP20
515  *   DESC:      Generated SAS Datasheet Code
516  *   TEMPLATE SOURCE: (None Specified.)
517  *****/
517! *****/
518  data BUAN6337.PIZZA ;
519  %let _EFIERR_ = 0; /* set the ERROR detection macro variable
519! */
520  infile 'Pizza.csv' delimiter = ',' MISOVER DSD lrecl=32767
520! firstobs=2 ;
521  informat SurveyNum best32. ;
522  informat Arugula best32. ;
523  informat PineNut best32. ;
524  informat Squash best32. ;
525  informat Shrimp $1. ;
526  informat Eggplant $1. ;
527  format SurveyNum best12. ;
528  format Arugula best12. ;
529  format PineNut best12. ;
530  format Squash best12. ;
531  format Shrimp $1. ;
532  format Eggplant $1. ;
533  input
534  SurveyNum
535  Arugula
536  PineNut
537  Squash
538  Shrimp $
539  Eggplant $
540  ;
541  if _ERROR_ then call symputx('_EFIERR_',1); /* set ERROR
541! detection macro variable */
542  run;

```


- d. Read the same raw data file, **Pizza.csv**, this time using a **DATA** step (instead of the **IMPORT** procedure). Be sure to resolve any issues identified above.

I use DATA step to informat/format each variable which ensures each variable change into *Num*.

```
5 data Buan6337.pizza2;
6 infile 'Pizza.csv' delimiter = ',' MISSOVER DSD lrecl=32767 firstobs=2 ;
7 informat SurveyNum best32. ;
8 informat Arugula best32. ;
9 informat PineNut best32. ;
10 informat Squash best32. ;
11 informat Shrimp best32. ;
12 informat Eggplant best32. ;
13 format SurveyNum best12. ;
14 format Arugula best12. ;
15 format PineNut best12. ;
16 format Squash best12. ;
17 format Shrimp best12. ;
18 format Eggplant best12. ;
19 input SurveyNum Arugula PineNut Squash Shrimp Eggplant;
20 run;
```

Print a report of the contents, and the result is change, this time type of variables is *Num*.

Alphabetic List of Variables and Attributes					
#	Variable	Type	Len	Format	Informat
2	Arugula	Num	8	BEST12.	BEST32.
6	Eggplant	Num	8	BEST12.	BEST32.
3	PineNut	Num	8	BEST12.	BEST32.
5	Shrimp	Num	8	BEST12.	BEST32.
4	Squash	Num	8	BEST12.	BEST32.
1	SurveyNum	Num	8	BEST12.	BEST32.

- e. Create a new dataset with the average ratings for each topping.

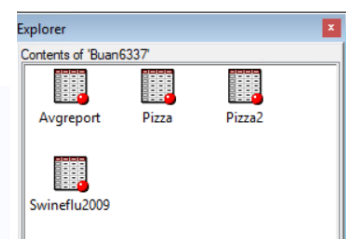
Coding part

```
PROC MEANS Data=Buan6337.pizza2 Mean Noprint nway;
VAR Arugula PineNut Squash Shrimp Eggplant;
OUTPUT out = pizza2_Result (drop= _TYPE_ _FREQ_)
Mean = / autoname;
run;
```

```
PROC PRINT DATA=pizza2_Result;
title ' The Average Ratings for Each Topping ';
RUN;
```

Result & New dataset

The Average Ratings for Each Topping					
Obs	Arugula_Mean	PineNut_Mean	Squash_Mean	Shrimp_Mean	Eggplant_Mean
1	3.075	3.14	3.1625	2.966666667	2.86



Question 3

The new management of a local hotel decided to update their recently acquired (and very outdated) property by installing wireless Internet service for their guests. They are also considering updating their billing system because the method used by the previous owner seems faulty. In order to conduct a billing analysis, they would like some calculations about the guests who stayed with them during the first part of February (this was the first month after the change of ownership). The raw data file Hotel.dat contains variables with information on room number, number of guests, check-in month, day, year, check-out month, day, year, use of wireless Internet service, number of days of Internet use, room type, and room rate.

a. Examine the raw data file Hotel.dat and read it into SAS.

The left is the code for input Hotel.dat into SAS, and the right is the SAS result.

```
1 data Buan6337.Hotel;
2 infile 'Hotel.dat';
3 input Room_No $ 1-4
4       Guest_Num 7-11
5       check_in_month 12-14
6       check_in_day 15-17
7       check_in_year 18-25
8       check_out_month 26-28
9       check_out_day 29-32
10      check_out_year 33-40
11      Wifi_use $ 41-47
12      Wifi_day_Num 48-51
13      Room_type $ 52-66
14      Room_rate ;
15 run;
16
17 proc print data=Buan6337.Hotel;
18 run;
```

Obs	Room_No	Guest_Num	check_in_month	check_in_day	check_in_year	check_out_month	check_out_day	check_out_year	Wifi_use	Wifi_day_Num	Room_type	Room_rate
1	211	3	2	7	2014	2	11	2014	NO		Deluxe Suite	295
2	214	2	2	2	2014	2	12	2014	NO		Basic no view	75
3	216	4	2	2	2014	2	13	2014	NO		Suite	255
4	220	5	2	3	2014	2	12	2014	YES	2	Basic whiew	155
5	221	3	2	3	2014	2	12	2014	NO		Luxury	195
6	223	5	2	7	2014	2	13	2014	NO		Suite	255
7	238	4	1	31	2014	2	13	2014	YES	10	Basic white	
8	241	1	2	1	2014	2	13	2014	YES	3	Luxury	195
9	244	5	2	3	2014	2	12	2014	YES	9	Deluxe Suite	295
10	247	4	2	7	2014	2	11	2014	YES	4	Basic no view	75
11	248	4	2	8	2014	2	13	2014	YES	5	Basic no view	75
12	253	3	2	2	2014	2	12	2014	YES	7	Suite	255
13	255	5	2	8	2014	2	13	2014	NO		Basic whiew	155
14	270	1	2	4	2014	2	12	2014	NO		Deluxe Suite	295
15	272	2	2	2	2014	2	11	2014	YES	7	Suite	255
16	275	1	2	2	2014	2	10	2014	NO		Basic no view	75
17	276	5	2	10	2014	2	12	2014	NO		Basic whiew	155
18	281	1	2	9	2014	2	10	2014	NO		Basic whiew	155
19	283	1	2	8	2014	2	11	2014	NO		Deluxe Suite	295
20	288	3	2	2	2014	2	13	2014	NO		Suite	255
21	293	4	2	7	2014	2	11	2014	YES	4	Basic no view	75
22	294	5	2	1	2014	2	12	2014	YES	1	Basic whiew	155

b. Create date variables for the check-in and check-out dates, and format them to display as readable dates.

I use “CATX” function to create new variable CheckIn and CheckOut.

```
data Buan6337.report(drop=check_in_month check_in_day check_in_year check_out_month check_out_day check_out_year)
set Buan6337.Hotel;
  CheckIn = cats(check_in_month,'/',check_in_day,'/',check_in_year);
  CheckOut = cats(check_out_month,'/',check_out_day,'/',check_out_year);
run;
```

Now need to let SAS know that this is actually a date. So, I combine “INPUT” function and a proper INFORMAT to let SAS recognize the date. The left is the code for format and the right is results.

```
1 data Buan6337.Hotel2 (drop = CheckIn CheckOut);
2   set Buan6337.report;
3   Check_In_Date= input(CheckIn, anydtdte32.);
4   Check_out_Date= input(CheckOut, anydtdte32.);
5   Format Check_In_Date mmddyy10.;
6   Format Check_out_Date mmddyy10.;
7   run;
```

#	Variable	Type	Len	Format
7	Check_In_Date	Num	8	MMDDYY10.
8	Check_out_Date	Num	8	MMDDYY10.
2	Guest_Num	Num	8	
1	Room_No	Char	4	
6	Room_rate	Num	8	
5	Room_type	Char	15	
4	Wifi_day_Num	Num	8	
3	Wifi_use	Char	7	

- c. Create a variable that calculates the subtotal as the room rate times the number of days in the stay, plus a per person rate (\$10 per day for each person beyond one guest), plus an Internet service fee (\$9.95 for a one-time activation and \$4.95 per day of use).

```
data Buan6337.Hotel3;
  set Buan6337.Hotel2;
  Days_in_stay = Check_out_Date-Check_In_Date;
  if Guest_Num = 1 then do ;
    Person_rate = 0;
  end;
  else do;
    Person_rate = ((Guest_Num-1)*10*Days_in_stay);
  end;

  if Wifi_use = 'NO' then do;
    Wifi_service_fee = 0;
  end;
  else do;
    Wifi_service_fee = 9.95+4.95*Wifi_day_Num;
  end;
run;
```

- d. Create a variable that calculates the grand total as the subtotal plus sales tax at 7.75%. The result should be rounded to two decimal places.

```
data Hotel4;
  set Hotel3;
  Grand_total = (Person_rate+Wifi_service_fee)*1.0775;
run;

proc print data= Hotel4;
run;

data Hotel_4;
  set Hotel4;
  Grand_total = round(Grand_total,0.01);
run;
```

- e. View the resulting data set. The value for the grand total for room 211 is \$86.20.

The SAS System												
Obs	Room_No	Guest_Num	Wifi_use	Wifi_day_Num	Room_type	Room_rate	Check_In_Date	Check_out_Date	Days_in_stay	Person_rate	Wifi_service_fee	Grand_total
1	211	3	NO	.	Deluxe Suite	295	02/07/2014	02/11/2014	4	80	0.00	86.20
2	214	2	NO	.	Basic no view	75	02/02/2014	02/12/2014	10	100	0.00	107.75
3	216	4	NO	.	Suite	255	02/02/2014	02/13/2014	11	330	0.00	355.58
4	220	5	YES	2	Basic w/view	155	02/03/2014	02/12/2014	9	360	19.85	409.29
5	221	3	NO	.	Luxury	195	02/03/2014	02/12/2014	9	180	0.00	193.95
6	223	5	NO	.	Suite	255	02/07/2014	02/13/2014	6	240	0.00	258.60
7	238	4	YES	10	Basic w/view	.	01/31/2014	02/13/2014	13	390	59.45	484.28
8	241	1	YES	3	Luxury	195	02/01/2014	02/13/2014	12	0	24.80	26.72
9	244	5	YES	9	Deluxe Suite	295	02/03/2014	02/12/2014	9	360	54.50	446.62
10	247	4	YES	4	Basic no view	75	02/07/2014	02/11/2014	4	120	29.75	161.36
11	248	4	YES	5	Basic no view	75	02/08/2014	02/13/2014	5	150	34.70	199.01
12	253	3	YES	7	Suite	255	02/02/2014	02/12/2014	10	200	44.60	263.56
13	255	5	NO	.	Basic w/view	155	02/08/2014	02/13/2014	5	200	0.00	215.50
14	270	1	NO	.	Deluxe Suite	295	02/04/2014	02/12/2014	8	0	0.00	0.00
15	272	2	YES	7	Suite	255	02/02/2014	02/11/2014	9	90	44.60	145.03
16	275	1	NO	.	Basic no view	75	02/02/2014	02/10/2014	8	0	0.00	0.00
17	276	5	NO	.	Basic w/view	155	02/10/2014	02/12/2014	2	80	0.00	86.20
18	281	1	NO	.	Basic w/view	155	02/09/2014	02/10/2014	1	0	0.00	0.00
19	283	1	NO	.	Deluxe Suite	295	02/08/2014	02/11/2014	3	0	0.00	0.00
20	288	3	NO	.	Suite	255	02/02/2014	02/13/2014	11	220	0.00	237.05
21	293	4	YES	4	Basic no view	75	02/07/2014	02/11/2014	4	120	29.75	161.36
22	294	5	YES	3	Basic w/view	155	02/01/2014	02/12/2014	11	440	24.80	500.82