Week 3 Task

Q1.) Import dataset in the SAS environment and check top 10 record of import dataset

```
FILENAME REFFILE '/home/u49809446/sasuser.v94/Life+Insurance+Dataset.csv';

PROC IMPORT DATAFILE=REFFILE

DBMS=CSV

OUT=sasuser.life_insurance;

GETNAMES=YES;

RUN;

 proc print data=sasuser.life_insurance(obs=10);

 run;
```

Obs	CustID	Mobile_num	Churn	Age	Payment_Period	Product	Cust_Tenure	EducationField	Gender	Overall_cust_satisfation_score	Cust_Designation	CC_Satisfation_score	Cust_MaritalStatus	Cust_Income	Agent_Tenure	Complaint	YTD_contact_cnt	Due_date_day_cnt
1	10002	9926913118	0	44	Monthly	Traditional	22	Statistics	Male	2	Manager	4	Divorced	20130	1	0	28	10
2	10005	9955950910	0	46	Yearly	Traditional	11	CA	Male	3	Executive	4	Divorced	18468	9	0	17	6
3	10009	9932307506	0	42	Monthly	Traditional	4	Statistics	Male	3	Senior Manager	3	Single	24526	0	0	26	10
4	10010	9879153854	0	43	Yearly	Traditional	23	CA	Male	5	Manager	3	Divorced	20237	6	0	18	17
5	10014	9885137899	0	50	Yearly	Traditional	19	CA	Male	5	Executive	2	Married	17661	0	0	16	3
6	10019	9918893968	0	43	Yearly	Pure Term Plan	19	Statistics	Female	2	AVP	2	Divorced	30427	2	0	21	31
7	10020	9880627494	0	39	Yearly	Traditional	15	Statistics	Male	3	Executive	2	Single	18944	5	1	16	6
8	10021	9952270464	0	32	Quarterly	Traditional	15	Other	Female	4	Manager	3	Married	19011	0	0	23	5
9	10022	9893757229	1	35	Yearly	Pure Term Plan	4	Statistics	Male	2	Manager	5	Single	18407	7	0	28	10
10	10026	9930780130	0	51	Yearly	Traditional	4	Other	Female	4	VP	3	Married	34094	4	0	16	26

Q2.) Check variable type of the import dataset

```
proc contents data=sasuser.life_insurance varnum;
run;
```

Variables in Creation Order

#	Variable	Type	Len	Format	Informat
1	CustID	Num	8	BEST12.	BEST32.
2	Mobile_num	Num	8	BEST12.	BEST32.
3	Churn	Num	8	BEST12.	BEST32.
4	Age	Num	8	BEST12.	BEST32.
5	Payment_Period	Char	9	$9.	$9.
6	Product	Char	14	$14.	$14.
7	Cust_Tenure	Num	8	BEST12.	BEST32.
8	EducationField	Char	17	$17.	$17.
9	Gender	Char	6	$6.	$6.
10	Overall_cust_satisfation_score	Num	8	BEST12.	BEST32.
11	Cust_Designation	Char	14	$14.	$14.
12	CC_Satisfation_score	Num	8	BEST12.	BEST32.
13	Cust_MaritalStatus	Char	8	$8.	$8.
14	Cust_Income	Num	8	BEST12.	BEST32.
15	Agent_Tenure	Num	8	BEST12.	BEST32.
16	Complaint	Num	8	BEST12.	BEST32.
17	YTD_contact_cnt	Num	8	BEST12.	BEST32.
18	Due_date_day_cnt	Num	8	BEST12.	BEST32.
19	Existing_policy_count	Num	8	BEST12.	BEST32.
20	Miss_due_date_cnt	Num	8	BEST12.	BEST32.

Q3.) Checks if any variables have missing values, if yes then do treatment?

```
proc means data=sasuser.life_insurance nmiss;
run;
```

The MEANS Procedure

Variable	N Miss
CustID	0
Mobile_num	0
Churn	0
Age	0
Cust_Tenure	0
Overall_cust_satisfation_score	0
CC_Satisfation_score	0
Cust_Income	0
Agent_Tenure	0
Complaint	0
YTD_contact_cnt	0
Due_date_day_cnt	0
Existing_policy_count	0
Miss_due_date_cnt	0

Q4.) Check summary and percentile distribution of all numerical variables for churners and non-churners?

```
proc means data=sasuser.Life_Insurance n nmiss min p1 p5 p10 p25 p50 p75 p90 p95 p99
max stddev Q1 Q3;
Class Churn;
Run;
```

Churn	N Obs	Variable	N	N Miss	Minimum	1st Pctl	5th Pctl	10th Pctl	25th Pctl	50th Pctl	75th Pctl	90th Pctl	95th Pctl	99th Pctl	Maximum	Std Dev	Lower Quartile	Upper Quartile
0	1607	CustID	1607	0	10002.00	10057.00	10325.00	10584.00	11422.00	12939.00	14436.00	15272.00	15576.00	15831.00	15878.00	1705.44	11422.00	14436.00
		Mobile_num	1607	0	9856826276	9857734052	9861557904	9867000851	9881411217	9905201357	9931145502	9945447387	9951416711	9955494492	9956738681	28679915.14	9881411217	9931145502
		Age	1607	0	30.0000000	30.0000000	31.0000000	33.0000000	37.0000000	45.0000000	53.0000000	58.0000000	59.0000000	60.0000000	60.0000000	8.8976782	37.0000000	53.0000000
		Cust_Tenure	1607	0	3.0000000	3.0000000	4.0000000	5.0000000	8.0000000	14.0000000	20.0000000	23.0000000	24.0000000	25.0000000	25.0000000	6.6591047	8.0000000	20.0000000
		Overall_cust_satisfation_score	1607	0	2.0000000	2.0000000	2.0000000	2.0000000	3.0000000	4.0000000	5.0000000	5.0000000	5.0000000	5.0000000	5.0000000	1.1242696	3.0000000	5.0000000
		CC_Satisfation_score	1607	0	1.0000000	1.0000000	1.0000000	1.0000000	2.0000000	3.0000000	4.0000000	5.0000000	5.0000000	5.0000000	5.0000000	1.3601511	2.0000000	4.0000000
		Cust_Income	1607	0	16051.00	17052.00	17345.00	17789.00	18760.00	20639.00	24268.00	31124.00	33665.00	35436.00	96000.00	5476.13	18760.00	24268.00
		Agent_Tenure	1607	0	0	0	0	1.0000000	1.0000000	2.0000000	4.0000000	7.0000000	8.0000000	10.0000000	10.0000000	2.4699358	1.0000000	4.0000000
		Complaint	1607	0	0	0	0	0	0	0	0	1.0000000	1.0000000	1.0000000	1.0000000	0.4269581	0	0
		YTD_contact_cnt	1607	0	16.0000000	16.0000000	16.0000000	17.0000000	18.0000000	20.0000000	23.0000000	26.0000000	28.0000000	30.0000000	31.0000000	3.6081023	18.0000000	23.0000000
		Due_date_day_cnt	1607	0	0	1.0000000	3.0000000	4.0000000	7.0000000	10.0000000	16.0000000	24.0000000	28.0000000	34.0000000	38.0000000	7.5568115	7.0000000	16.0000000
		Existing_policy_count	1607	0	1.0000000	1.0000000	1.0000000	2.0000000	4.0000000	8.0000000	12.0000000	14.0000000	15.0000000	15.0000000	15.0000000	4.3329530	4.0000000	12.0000000
		Miss_due_date_cnt	1607	0	0	0	0	0	0	1.0000000	2.0000000	2.0000000	2.0000000	2.0000000	2.0000000	0.8148688	0	2.0000000
1	317	CustID	317	0	10022.00	10108.00	10289.00	10592.00	11291.00	12824.00	14273.00	15377.00	15673.00	15776.00	15872.00	1716.33	11291.00	14273.00
		Mobile_num	317	0	9856860057	9857927009	9859458007	9864239662	9877795809	9902041357	9929488928	9948092373	9952581357	9955625914	9956330829	29636261.03	9877795809	9929488928
		Age	317	0	21.0000000	21.0000000	21.0000000	23.0000000	25.0000000	31.0000000	36.0000000	38.0000000	39.0000000	40.0000000	40.0000000	5.6710337	25.0000000	36.0000000
		Cust_Tenure	317	0	1.0000000	1.0000000	1.0000000	1.0000000	3.0000000	5.0000000	8.0000000	9.0000000	10.0000000	10.0000000	10.0000000	2.8831722	3.0000000	8.0000000
		Overall_cust_satisfation_score	317	0	1.0000000	1.0000000	1.0000000	1.0000000	2.0000000	3.0000000	4.0000000	4.0000000	4.0000000	4.0000000	4.0000000	1.1052958	2.0000000	4.0000000
		CC_Satisfation_score	317	0	1.0000000	1.0000000	1.0000000	1.0000000	3.0000000	4.0000000	5.0000000	5.0000000	5.0000000	5.0000000	5.0000000	1.3159034	3.0000000	5.0000000
		Cust_Income	317	0	16009.00	16261.00	17044.00	17275.00	17693.00	18691.00	21381.00	25325.00	26609.00	34545.00	35859.00	3547.15	17693.00	21381.00
		Agent_Tenure	317	0	0	0	1.0000000	1.0000000	1.0000000	2.0000000	5.0000000	8.0000000	9.0000000	10.0000000	10.0000000	2.6466454	1.0000000	5.0000000
		Complaint	317	0	0	0	0	0	0	1.0000000	1.0000000	1.0000000	1.0000000	1.0000000	1.0000000	0.4992307	0	1.0000000
		YTD_contact_cnt	317	0	16.0000000	16.0000000	16.0000000	17.0000000	18.0000000	20.0000000	23.0000000	27.0000000	28.0000000	30.0000000	30.0000000	3.7855715	18.0000000	23.0000000
		Due_date_day_cnt	317	0	0	1.0000000	1.0000000	2.0000000	4.0000000	8.0000000	11.0000000	18.0000000	24.0000000	33.0000000	41.0000000	6.9856842	4.0000000	11.0000000
		Existing_policy_count	317	0	1.0000000	1.0000000	1.0000000	2.0000000	4.0000000	8.0000000	12.0000000	14.0000000	15.0000000	15.0000000	15.0000000	4.3015897	4.0000000	12.0000000
		Miss_due_date_cnt	317	0	2.0000000	2.0000000	2.0000000	2.0000000	4.0000000	6.0000000	8.0000000	10.0000000	10.0000000	10.0000000	10.0000000	2.5934324	4.0000000	8.0000000

Q5.) Check for outlier, if yes then do treatment?

```
proc univariate data=sasuser.life_insurance;
 var Age Cust_Tenure Overall_cust_satisfation_score CC_Satisfation_score Cust_Income
Agent_Tenure YTD_contact_cnt Due_date_day_cnt Existing_policy_count Miss_due_date_cnt;
  run;
  data life_insurance;
  set sasuser.life_insurance;
  if Cust_Tenure > 37 then Cust_Tenure = 37;
  if Cust_Income >31585 then Cust_Income = 31585;
  if Cust_Income <10738 then Cust_Income = 10738;
  if Miss_due_date_cnt > 5 then Miss_due_date_cnt = 5;
  if Due_date_day_cnt > 29.75 then Due_date_day_cnt = 30;
  if YTD_contact_cnt >30 then YTD_contact_cnt=30;
  run;
```

The UNIVARIATE Procedure
Variable: Age

Moments			
N	1924	Sum Weights	1924
Mean	42.6242204	Sum Observations	82009
Std Deviation	10.0113121	Variance	100.226371
Skewness	0.00576962	Kurtosis	-0.9543936
Uncorrected SS	3688305	Corrected SS	192735.311
Coeff Variation	23.4873789	Std Error Mean	0.22823827

Basic Statistical Measures			
Location		Variability	
Mean	42.62422	Std Deviation	10.01131
Median	42.00000	Variance	100.22637
Mode	38.00000	Range	39.00000
		Interquartile Range	17.00000

Tests for Location: Mu0=0				
Test	Statistic		p Value	
Student's t	t	186.7532	Pr > |t|	<.0001
Sign	M	962	Pr >= |M|	<.0001
Signed Rank	S	925925	Pr >= |S|	<.0001

Quantiles (Definition 5)	
Level	Quantile
100% Max	60
99%	60
95%	59
90%	57
75% Q3	51
50% Median	42
25% Q1	34
10%	30
5%	27
1%	22
0% Min	21

Q6.) Check the proportion of all categorical variables and extract percentage contribution of each class in respective variables?

```
proc freq data=life_insurance;
table Churn Payment_Period Product EducationField Gender Cust_Designation
Cust_MaritalStatus /nocum;
run;
```

The FREQ Procedure

Churn	Frequency	Percent
0	1607	83.52
1	317	16.48

Payment_Period	Frequency	Percent
Monthly	345	17.93
Quarterly	189	9.82
Yearly	1390	72.25

Product	Frequency	Percent
Market Link	81	4.21
Pure Term Plan	560	29.11
Traditional	1283	66.68

EducationField	Frequency	Percent
CA	583	30.30
Engineer	188	9.77
MBA	30	1.56
Marketing Diploma	219	11.38
Other	110	5.72
Statistics	794	41.27

Gender	Frequency	Percent
Female	732	38.05
Male	1192	61.95

Cust_Designation	Frequency	Percent
AVP	139	7.22
Executive	723	37.58
Manager	679	35.29
Senior Manager	298	15.49
VP	85	4.42

Q7.) Customer service management want you to create a macro where they will just put mobile number and they will get all the important information like Age, Education, Gender, Income and CustID

```
%MACRO cust_info();
DATA output (keep = Mobile_num Age EducationField Gender Cust_Income CustID);
SET life_insurance;
where Mobile_num in (&Mobile_num.);
RUN;

proc print data=output;
run;
%MEND;
```

```
%let Mobile_num = 9926913118;
%cust_info;
```

Obs	CustID	Mobile_num	Age	EducationField	Gender	Cust_Income
1	10002	9926913118	44	Statistics	Male	20130

Q8.) Check correlation of all numerical variables before building model, because we cannot add correlated variables in model?

```
proc corr data=life_insurance noprob;
var Age Cust_Tenure Overall_cust_satisfation_score CC_Satisfation_score Cust_Income
Agent_Tenure YTD_contact_cnt Due_date_day_cnt Existing_policy_count
Miss_due_date_cnt;
run;
```

The CORR Procedure

10 Variables: Age Cust_Tenure Overall_cust_satisfation_score CC_Satisfation_score Cust_Income Agent_Tenure YTD_contact_cnt Due_date_day_cnt Existing_policy_count Miss_due_date_cnt

Simple Statistics

Variable	N	Mean	Std Dev	Sum	Minimum	Maximum
Age	1924	42.62422	10.01131	82009	21.00000	60.00000
Cust_Tenure	1924	12.64865	7.01534	24336	1.00000	25.00000
Overall_cust_satisfation_score	1924	3.39553	1.18053	6533	1.00000	5.00000
CC_Satisfation_score	1924	3.05146	1.36632	5871	1.00000	5.00000
Cust_Income	1924	21786	4286	41915503	16009	31585
Agent_Tenure	1924	3.16320	2.50125	6086	0	10.00000
YTD_contact_cnt	1924	20.66476	3.62567	39759	16.00000	30.00000
Due_date_day_cnt	1924	11.55353	7.29647	22229	0	30.00000
Existing_policy_count	1924	8.09304	4.32749	15571	1.00000	15.00000
Miss_due_date_cnt	1924	1.53170	1.51145	2947	0	5.00000

Pearson Correlation Coefficients, N = 1924

	Age	Cust_Tenure	Overall_cust_satisfation_score	CC_Satisfation_score	Cust_Income	Agent_Tenure	YTD_contact_cnt	Due_date_day_cnt	Existing_policy_count	Miss_due_date_cnt
Age	1.00000	0.26821	0.18660	-0.11028	0.06980	-0.03420	-0.00108	0.09127	0.01701	-0.43661
Cust_Tenure	0.26821	1.00000	0.17760	-0.07049	0.07384	-0.01564	0.02939	0.08258	0.01271	-0.37526
Overall_cust_satisfation_score	0.18660	0.17760	1.00000	-0.05454	0.07062	-0.03825	-0.01590	0.03874	0.00643	-0.25636
CC_Satisfation_score	-0.11028	-0.07049	-0.05454	1.00000	0.01895	0.07712	0.00527	0.02740	0.02944	0.12625
Cust_Income	0.06980	0.07384	0.07062	0.01895	1.00000	0.19415	0.00097	0.77264	0.02375	-0.13995
Agent_Tenure	-0.03420	-0.01564	-0.03825	0.07712	0.19415	1.00000	0.02611	0.26127	-0.00866	0.02958
YTD_contact_cnt	-0.00108	0.02939	-0.01590	0.00527	0.00097	0.02611	1.00000	0.00686	0.05094	-0.01699
Due_date_day_cnt	0.09127	0.08258	0.03874	0.02740	0.77264	0.26127	0.00686	1.00000	0.03251	-0.14124
Existing_policy_count	0.01701	0.01271	0.00643	0.02944	0.02375	-0.00866	0.05094	0.03251	1.00000	0.00325
Miss_due_date_cnt	-0.43661	-0.37526	-0.25636	0.12625	-0.13995	0.02958	-0.01699	-0.14124	0.00325	1.00000

Q9.) Create train and test (70:30) dataset from the existing data set. Put seed 1234?

```
proc freq data=life_insurance;
table Churn /nocum;
run;

proc surveyselect data=life_insurance rate=0.3  seed=1234
out=test
method=srs;
run;

proc contents data=test varnum;
run;

proc freq data=test;
table Churn /nocum;
run;
```

```
proc sql;
create table train as select t1.* from life_insurance as t1
where Mobile_num not in (select Mobile_num from test);
quit;

proc freq data=train;
table Churn /nocum;
run;
```

The CONTENTS Procedure

Data Set Name	WORK.TEST	Observations	578
Member Type	DATA	Variables	21
Engine	V9	Indexes	0
Created	10/02/2020 15:01:05	Observation Length	192
Last Modified	10/02/2020 15:01:05	Deleted Observations	0
Protection		Compressed	NO
Data Set Type		Sorted	NO
Label			
Data Representation	SOLARIS_X86_64, LINUX_X86_64, ALPHA_TRU64, LINUX_IA64		
Encoding	utf-8 Unicode (UTF-8)		

TEST SET (30% DATA)

The CONTENTS Procedure

Data Set Name	WORK.TRAIN	Observations	1346
Member Type	DATA	Variables	21
Engine	V9	Indexes	0
Created	10/02/2020 15:01:25	Observation Length	192
Last Modified	10/02/2020 15:01:25	Deleted Observations	0
Protection		Compressed	NO
Data Set Type		Sorted	NO
Label			
Data Representation	SOLARIS_X86_64, LINUX_X86_64, ALPHA_TRU64, LINUX_IA64		
Encoding	utf-8 Unicode (UTF-8)		

TRAINING SET (70% DATA)

Q10.) Develop linear regression model first on the target variable to extract VIF information to check multicollinearity?

```
proc reg data=train;
model churn=Age Cust_Tenure Overall_cust_satisfation_score CC_Satisfation_score
Cust_Income Agent_Tenure Complaint YTD_contact_cnt  Existing_policy_count
Miss_due_date_cnt /vif tol;
run;
```

The REG Procedure
Model: MODEL1
Dependent Variable: Churn

Number of Observations Read	1346
Number of Observations Used	1346

Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	10	135.34658	13.53466	376.27	<.0001
Error	1335	48.02118	0.03597		
Corrected Total	1345	183.36776			

Root MSE	0.18966	R-Square	0.7381
Dependent Mean	0.16270	Adj R-Sq	0.7362
Coeff Var	116.56730		

Parameter Estimates							
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Tolerance	Variance Inflation
Intercept	1	0.45290	0.05274	8.59	<.0001	.	0
Age	1	-0.00688	0.00057768	-11.91	<.0001	0.78657	1.27134
Cust_Tenure	1	-0.00738	0.00081846	-9.01	<.0001	0.83325	1.20012
Overall_cust_satisfation_score	1	-0.02518	0.00460	-5.48	<.0001	0.91921	1.08789
CC_Satisfation_score	1	0.00983	0.00387	2.54	0.0112	0.96514	1.03612
Cust_Income	1	-0.00000493	0.00000122	-4.04	<.0001	0.93345	1.07130
Agent_Tenure	1	0.00375	0.00213	1.76	0.0785	0.94918	1.05354
Complaint	1	0.05260	0.01166	4.51	<.0001	0.94794	1.05492
YTD_contact_cnt	1	0.00050852	0.00143	0.35	0.7229	0.99313	1.00692
Existing_policy_count	1	-0.00187	0.00119	-1.57	0.1173	0.99539	1.00463
Miss_due_date_cnt	1	0.15607	0.00415	37.58	<.0001	0.68604	1.45765

Q11.) Create clean logistic model on the target variables?

```
proc logistic data= train;
model Churn=Age Cust_Tenure Overall_cust_satisfation_score CC_Satisfation_score
Cust_Income Agent_Tenure Complaint YTD_contact_cnt Existing_policy_count
Miss_due_date_cnt /lackfit;
output out = train_output xbeta = coeff stdxbeta = stdcoeff predicted = prob;
run;
```

Analysis of Maximum Likelihood Estimates					
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	6.9377	104.0	0.0045	0.9468
Age	1	0.3217	0.0834	14.8926	0.0001
Cust_Tenure	1	0.4912	0.1297	14.3543	0.0002
Overall_cust_satisfa	1	0.9813	0.3572	7.5461	0.0060
CC_Satisfation_score	1	-0.8476	0.3447	6.0469	0.0139
Cust_Income	1	0.000060	0.000079	0.5736	0.4488
Agent_Tenure	1	-0.1320	0.1595	0.6850	0.4079
Complaint	1	-2.6403	0.9100	8.4176	0.0037
YTD_contact_cnt	1	0.2583	0.1245	4.3066	0.0380
Existing_policy_coun	1	0.1386	0.0852	2.6503	0.1035
Miss_due_date_cnt	1	-13.1507	51.9484	0.0641	0.8002

Odds Ratio Estimates			
Effect	Point Estimate	95% Wald Confidence Limits	
Age	1.380	1.172	1.624
Cust_Tenure	1.634	1.268	2.107
Overall_cust_satisfa	2.668	1.325	5.373
CC_Satisfation_score	0.428	0.218	0.842
Cust_Income	1.000	1.000	1.000
Agent_Tenure	0.876	0.641	1.198
Complaint	0.071	0.012	0.425
YTD_contact_cnt	1.295	1.014	1.652
Existing_policy_coun	1.149	0.972	1.357
Miss_due_date_cnt	<0.001	<0.001	>999.999

Association of Predicted Probabilities and Observed Responses			
Percent Concordant	99.9	Somers' D	0.999
Percent Discordant	0.1	Gamma	0.999
Percent Tied	0.0	Tau-a	0.272
Pairs	246813	c	0.999

Inference→ From the table we can see that for columns Cust_Income,Agent_Tenure,Existing_policy_count and Miss_due_date_cnt the P- value is greater than 0.05(5%), which means that these variables are insignificant for our logistic model , so we can remove these variables from our model to increase accuracy.

After removing the new model looks like this:

Analysis of Maximum Likelihood Estimates					
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-13.3759	1.4426	85.9737	<.0001
Age	1	0.2932	0.0281	108.6437	<.0001
Cust_Tenure	1	0.3882	0.0430	81.4756	<.0001
Overall_cust_satisfa	1	0.8359	0.1315	40.3999	<.0001
CC_Satisfation_score	1	-0.3607	0.0989	13.2914	0.0003
Complaint	1	-0.9974	0.2800	12.6880	0.0004
YTD_contact_cnt	1	0.00657	0.0364	0.0326	0.8567

Now we can see the YTD_contact_cnt has also become insignificant for our model (P-value >0.05) hence we can further remove this from our model as well. So, finally our model after removing the insignificant variables would look like:

Model Fit Statistics		
Criterion	Intercept Only	Intercept and Covariates
AIC	1197.590	380.921
SC	1202.795	412.150
-2 Log L	1195.590	368.921

Testing Global Null Hypothesis: BETA=0			
Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	826.6693	5	<.0001
Score	604.8976	5	<.0001
Wald	173.4131	5	<.0001

Analysis of Maximum Likelihood Estimates					
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-13.2425	1.2352	114.9372	<.0001
Age	1	0.2933	0.0281	108.6793	<.0001
Cust_Tenure	1	0.3884	0.0430	81.5887	<.0001
Overall_cust_satisfa	1	0.8356	0.1315	40.3870	<.0001
CC_Satisfation_score	1	-0.3614	0.0989	13.3548	0.0003
Complaint	1	-0.9948	0.2796	12.6608	0.0004

Odds Ratio Estimates			
Effect	Point Estimate	95% Wald Confidence Limits	
Age	1.341	1.269	1.417
Cust_Tenure	1.475	1.355	1.604
Overall_cust_satisfa	2.306	1.782	2.984
CC_Satisfation_score	0.697	0.574	0.846
Complaint	0.370	0.214	0.640

Now we can see all the variables have a P-value less than 0.05 which means they are significant for our logistic model.

Q12.) Create a macro and take a KS approach to take a cut off on the calculated scores?

```
%let variables=Age Cust_Tenure Overall_cust_satisfation_score CC_Satisfation_score
Complaint  ;
 proc logistic data= train;
model Churn=&variables /lackfit;
```

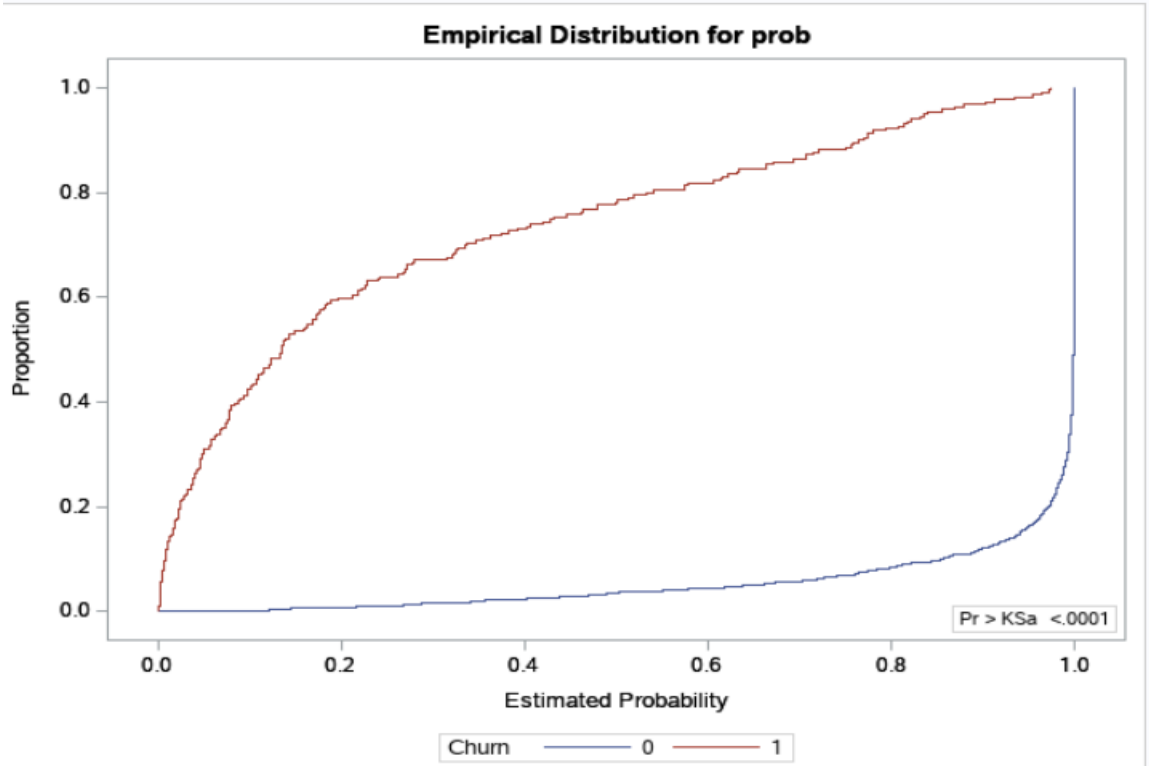
```
output out = train_output xbeta = coeff stdxbeta = stdcoeff predicted = prob;
run;

Proc npar1way data=train_output edf;
class Churn;
var prob;
run;
```

The NPAR1WAY Procedure

Kolmogorov-Smirnov Test for Variable prob Classified by Variable Churn			
Churn	N	EDF at Maximum	Deviation from Mean at Maximum
0	1127	0.098492	-4.699674
1	219	0.958904	10.661238
Total	1346	0.238484	
Maximum Deviation Occurred at Observation 1043			
Value of prob at Maximum = 0.854497			

Kolmogorov-Smirnov Two-Sample Test (Asymptotic)			
KS	0.317575	D	0.860413
KSa	11.651135	Pr > KSa	<.0001



Inference→ The 'D'value gives the indication of the KS value of the dataset. Higher the D value means better the model is able to distinguish between events and non-events (Churn = 1 or 0 in this case). The value comes out to be 0.86 which is good.

Q13.) Predict test dataset using created model?

```
%let variables=Age Cust_Tenure Overall_cust_satisfation_score CC_Satisfation_score
Complaint  ;
proc logistic data=train;
model Churn = &variables;
score data=test out=mypreds;
run;
```

```
/* Confusion matrix */
proc freq data=mypreds;
tables F_Churn*I_Churn /NOROW NOCOL;
RUN;
```

The FREQ Procedure

Frequency Percent	Table of F_Churn by I_Churn		
	I_Churn(Into: Churn)		
F_Churn(From: Churn)	0	1	Total
0	469 81.14	11 1.90	480 83.04
1	19 3.29	79 13.67	98 16.96
Total	488 84.43	90 15.57	578 100.00

/*Confusion Matrix*/

Inference→ From the confusion matrix we can see that the model has a very high accuracy (approx. 95%) and only 30 incorrect predictions out of a total of 578 observations.
