Credit Card Fraud Detection Project

Yousef Waiel Said

1/22/2024

I. Introduction and Overview

The dataset contains transactions made by credit cards in September 2013 by card- holders in two-day period. Of 284,807 valid transactions, 492 are listed as fraudulent. The variable 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset. The variable 'Amount' is the transaction value. The variable 'Class' is the response variable where 1 is a case of fraud and 0 is a valid transaction.

II. Dataset and Exploratory Analysis

The dataset for this project can be downloaded here:

https://www.kaggle.com/mlg-ulb/creditcardfraud

First, we will examine the data and provide any initial conclusions.

The number of rows in the dataset:

```
## [1] 284807
```

The number of columns in the dataset:

```
## [1] 31
```

We can see the first six full entries of the dataset:

```
##   Time         V1          V2        V3         V4          V5          V6
## 1    0 -1.3598071 -0.07278117 2.5363467  1.3781552 -0.33832077  0.46238778
## 2    0  1.1918571  0.26615071 0.1664801  0.4481541  0.06001765 -0.08236081
## 3    1 -1.3583541 -1.34016307 1.7732093  0.3797796 -0.50319813  1.80049938
## 4    1 -0.9662717 -0.18522601 1.7929933 -0.8632913 -0.01030888  1.24720317
## 5    2 -1.1582331  0.87773675 1.5487178  0.4030339 -0.40719338  0.09592146
## 6    2 -0.4259659  0.96052304 1.1411093 -0.1682521  0.42098688 -0.02972755
##            V7          V8         V9         V10        V11         V12
```

```
## 1  0.23959855  0.09869790  0.3637870  0.09079417 -0.5515995 -0.61780086
## 2 -0.07880298  0.08510165 -0.2554251 -0.16697441  1.6127267  1.06523531
## 3  0.79146096  0.24767579 -1.5146543  0.20764287  0.6245015  0.06608369
## 4  0.23760894  0.37743587 -1.3870241 -0.05495192 -0.2264873  0.17822823
## 5  0.59294075 -0.27053268  0.8177393  0.75307443 -0.8228429  0.53819555
## 6  0.47620095  0.26031433 -0.5686714 -0.37140720  1.3412620  0.35989384
##          V13        V14        V15        V16         V17         V18
## 1 -0.9913898 -0.3111694  1.4681770 -0.4704005  0.20797124  0.02579058
## 2  0.4890950 -0.1437723  0.6355581  0.4639170 -0.11480466 -0.18336127
## 3  0.7172927 -0.1659459  2.3458649 -2.8900832  1.10996938 -0.12135931
## 4  0.5077569 -0.2879237 -0.6314181 -1.0596472 -0.68409279  1.96577500
## 5  1.3458516 -1.1196698  0.1751211 -0.4514492 -0.23703324 -0.03819479
## 6 -0.3580907 -0.1371337  0.5176168  0.4017259 -0.05813282  0.06865315
##           V19         V20          V21          V22         V23         V24
## 1  0.40399296  0.25141210 -0.018306778  0.277837576 -0.11047391  0.06692807
## 2 -0.14578304 -0.06908314 -0.225775248 -0.638671953  0.10128802 -0.33984648
## 3 -2.26185710  0.52497973  0.247998153  0.771679402  0.90941226 -0.68928096
## 4 -1.23262197 -0.20803778 -0.108300452  0.005273597 -0.19032052 -1.17557533
## 5  0.80348692  0.40854236 -0.009430697  0.798278495 -0.13745808  0.14126698
## 6 -0.03319379  0.08496767 -0.208253515 -0.559824796 -0.02639767 -0.37142658
##          V25        V26          V27         V28 Amount Class
## 1  0.1285394 -0.1891148  0.133558377 -0.02105305 149.62     0
## 2  0.1671704  0.1258945 -0.008983099  0.01472417   2.69     0
## 3 -0.3276418 -0.1390966 -0.055352794 -0.05975184 378.66     0
## 4  0.6473760 -0.2219288  0.062722849  0.06145763 123.50     0
## 5 -0.2060096  0.5022922  0.219422230  0.21515315  69.99     0
## 6 -0.2327938  0.1059148  0.253844225  0.08108026   3.67     0
```

To better understand the data we present a data dictionary of the 31 variables in the dataset.

- **Time** - the number of seconds elapsed between this transaction and the first transaction in the dataset
- **V1-V28** is the result of a PCA Dimensionality reduction to protect user identities and sensitive features
- **Amount** - the dollar value of the transaction
- **Class** - 1 for fraudulent transactions, 0 for valid transactions

Implementing the variable header to the left column gives us another method to observe the first few entries of the data collection. We can additionally see that the collection has 31 variables totaling 284,807 entries.

```
## Rows: 284,807
## Columns: 31
## $ Time   <dbl> 0, 0, 1, 1, 2, 2, 4, 7, 7, 9, 10, 10, 10, 11, 12, 12, 12, 13, 1~
## $ V1     <dbl> -1.3598071, 1.1918571, -1.3583541, -0.9662717, -1.1582331, -0.4~
## $ V2     <dbl> -0.07278117, 0.26615071, -1.34016307, -0.18522601, 0.87773675, ~
## $ V3     <dbl> 2.53634674, 0.16648011, 1.77320934, 1.79299334, 1.54871785, 1.1~
## $ V4     <dbl> 1.37815522, 0.44815408, 0.37977959, -0.86329128, 0.40303393, -0~
## $ V5     <dbl> -0.33832077, 0.06001765, -0.50319813, -0.01030888, -0.40719338,~
## $ V6     <dbl> 0.46238778, -0.08236081, 1.80049938, 1.24720317, 0.09592146, -0~
## $ V7     <dbl> 0.239598554, -0.078802983, 0.791460956, 0.237608940, 0.59294074~
## $ V8     <dbl> 0.098697901, 0.085101655, 0.247675787, 0.377435875, -0.27053267~
## $ V9     <dbl> 0.3637870, -0.2554251, -1.5146543, -1.3870241, 0.8177393, -0.56~
## $ V10    <dbl> 0.09079417, -0.16697441, 0.20764287, -0.05495192, 0.75307443, -~
## $ V11    <dbl> -0.55159953, 1.61272666, 0.62450146, -0.22648726, -0.82284288, ~
## $ V12    <dbl> -0.61780086, 1.06523531, 0.06608369, 0.17822823, 0.53819555, 0.~
## $ V13    <dbl> -0.99138985, 0.48909502, 0.71729273, 0.50775687, 1.34585159, -0~
```

Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	Amount	Class
0	-1.3598071	-0.0727812	2.5363467	1.3781552	-0.3383208	0.4623878	0.2395986	0.0986979	0.3637870	0.0907942	-0.5515995	-0.6178009	-0.9913898	-0.3111694	1.4681770	-0.4704005	0.2079712	0.0257906	0.4039930	0.2514121	-0.0183068	0.2778376	-0.1104739	0.0669281	0.1285394	-0.1891148	0.1335584	-0.0210531	149.62	0
0	1.1918571	0.2661507	0.1664801	0.4481541	0.0600176	-0.0823608	-0.0780030	0.0851017	-0.2554251	-0.1669744	1.6125267	1.0652353	0.4899950	-0.1437723	0.6355581	0.4639170	-0.1148047	-0.1833613	-0.1457830	-0.0690831	-0.2257752	-0.6386720	0.1012880	-0.3393465	0.1671704	0.1258945	-0.0089831	0.0147242	2.69	0
1	-1.3583541	-1.3401631	1.7732093	0.3797796	-0.5031981	1.8004994	0.7914610	0.2476758	-1.5146543	0.2076429	0.6245015	0.0660837	0.7172927	-0.1659459	2.3458649	-2.8900832	1.1099604	-0.1213593	-2.2818571	0.5248797	0.2479982	0.7716794	0.9094123	-0.6892810	-0.3276418	-0.1390966	-0.0553528	-0.0597518	378.66	0
1	-0.9662717	-0.1852300	1.7929933	-0.8632913	-0.0103089	1.2472032	0.2376089	0.3774359	-1.3870241	-0.0549519	-0.2264873	0.1782282	0.5071569	-0.2879237	-0.6314181	-1.0596472	-0.6840928	1.9657750	-1.2326220	-0.3089378	-0.1083895	0.0852596	-0.1903205	-1.1755753	0.6473769	-0.2219288	0.0627228	0.0614576	123.50	0
2	-1.1582331	0.8777368	1.5487178	0.4030339	-0.4071934	0.0959215	0.5929407	-0.2705327	0.8177393	0.7530744	-0.8228429	0.5381956	1.3458516	-1.1196698	0.1751211	-0.4514492	-0.2705332	-0.0381948	0.8034869	0.4085424	-0.0094307	0.7982785	-0.1374581	0.1412670	-0.2660696	0.5022922	0.2194222	0.2151531	69.99	0
2	-0.4259659	0.9605230	1.1411093	-0.1682521	0.4209960	0.0392726	0.4762069	0.2604143	-0.5698714	0.3714972	1.3412620	0.3590938	-0.3589907	-0.1371337	0.5176168	0.4017259	-0.0581328	0.0886531	-0.0331938	0.0849677	-0.3082535	-0.5593248	-0.0263972	-0.3714366	-0.2327938	0.1059148	0.2538442	0.0810803	3.67	0
4	1.2296576	0.1410035	0.0453708	1.2026127	0.1918810	0.2727081	-0.0655190	0.0812129	0.4646600	-0.0992543	-1.4169072	-0.1538258	-0.7510627	0.1673720	0.6501436	-0.4435868	0.0228205	-0.6119873	-0.0455570	-0.2196326	-0.1677163	-0.2707097	-0.1541038	-0.7800554	0.7501369	-0.2572368	0.0345974	0.0951678	4.99	0
7	-0.6442694	1.4179635	1.0743694	-0.4921990	0.9493941	0.4291185	1.1206314	-3.8076642	0.6153747	1.2493762	-0.6194878	0.2914744	1.7579642	-1.3238652	0.6861325	-0.0761270	-1.2221273	-0.3582216	0.3245947	-0.1567419	1.9434653	-1.0154547	0.6075035	-0.6497099	-0.4153666	-0.0516343	-1.2689211	-1.1853392	40.80	0
7	-0.8942861	0.2861572	-0.1131922	-0.2715261	2.6695987	3.7218181	0.3704151	0.8510844	-0.3929676	-0.4104304	-0.7051166	-0.1104523	-0.2862536	0.6743554	-0.8287831	-0.2100773	-0.4997880	0.1187849	0.5705282	0.0527357	-0.0734251	-0.2680916	-0.2042327	1.0111918	0.3723047	-0.3841573	0.0117474	0.1424943	93.20	0
9	-0.3382618	1.1195934	1.0443760	-0.2221873	0.4993608	-0.2467611	0.6515832	0.0695386	-0.7367273	-0.3668456	1.0176145	0.8303896	-0.4435228	0.1052191	0.7304528	-0.5409799	0.4766773	0.4517730	0.2037115	-0.2460139	-0.6337526	-0.1207941	-0.3850499	-0.0697530	0.0941983	0.2482193	0.0930756	3.68	0
10	1.4490439	-1.1763388	0.9138598	-1.3756667	-1.9713832	-0.6291521	-1.4232356	0.0144559	-1.7294084	1.6266591	1.1996439	-0.6714398	-0.5139472	-0.0950450	0.2269304	0.0319675	0.2534147	0.8543438	-0.2213654	-0.3872265	-0.0693019	0.3138944	0.0277402	0.5005123	0.2513674	-0.1294780	0.0428499	0.0162533	7.80	0
10	0.3849782	0.6161095	-0.8742997	-0.0940186	2.9245844	3.3170272	0.4704547	0.5382473	-0.5588946	0.3097554	-0.2591156	-0.3261432	-0.0600467	0.3628324	0.9289037	-0.1294868	-0.8099780	0.3599854	0.7076638	0.1259916	0.0499237	0.2384215	0.0691299	0.9967102	-0.7673148	-0.4922083	0.0424724	-0.0543374	9.99	0
10	1.2499867	1.2216368	0.3839302	-1.2348987	-1.4854195	-0.7532302	-0.6894050	-0.2274872	-2.0940106	1.3221293	0.2276662	-0.2426820	1.2054168	-0.3176305	0.7256750	-0.8156122	0.8739364	-0.8477886	-0.6831926	-0.1027559	-0.2318092	-0.4832853	0.0846677	0.3923499	0.1611346	-0.3549900	0.0266155	0.0424221	121.50	0
11	1.0693736	0.2877221	0.8288127	2.7125204	-0.1783990	0.3375437	-0.0987160	0.1159917	-0.2210826	0.4602304	-0.7736569	0.3233572	-0.0110959	-0.1284852	-0.6655643	-0.1399252	0.1240954	-0.9801062	-0.9829161	-0.1531972	-0.0368726	0.0744124	-0.0714955	0.1047438	0.5482842	0.1040942	0.0214911	0.0212583	27.50	0
12	-2.7918548	-0.3277708	1.6417592	1.7674727	-0.1365884	0.8075965	-0.4229114	-1.9071075	0.7557129	1.1510870	0.8445555	0.7929440	0.3704481	-0.7349751	0.4067957	-0.3030576	-0.1558687	0.7282655	2.2218680	-1.5821220	1.1516630	0.2221820	1.0205862	0.0283167	-0.2327463	-0.2355572	-0.1647775	-0.0301536	58.80	0

Length	Columns
284807	31

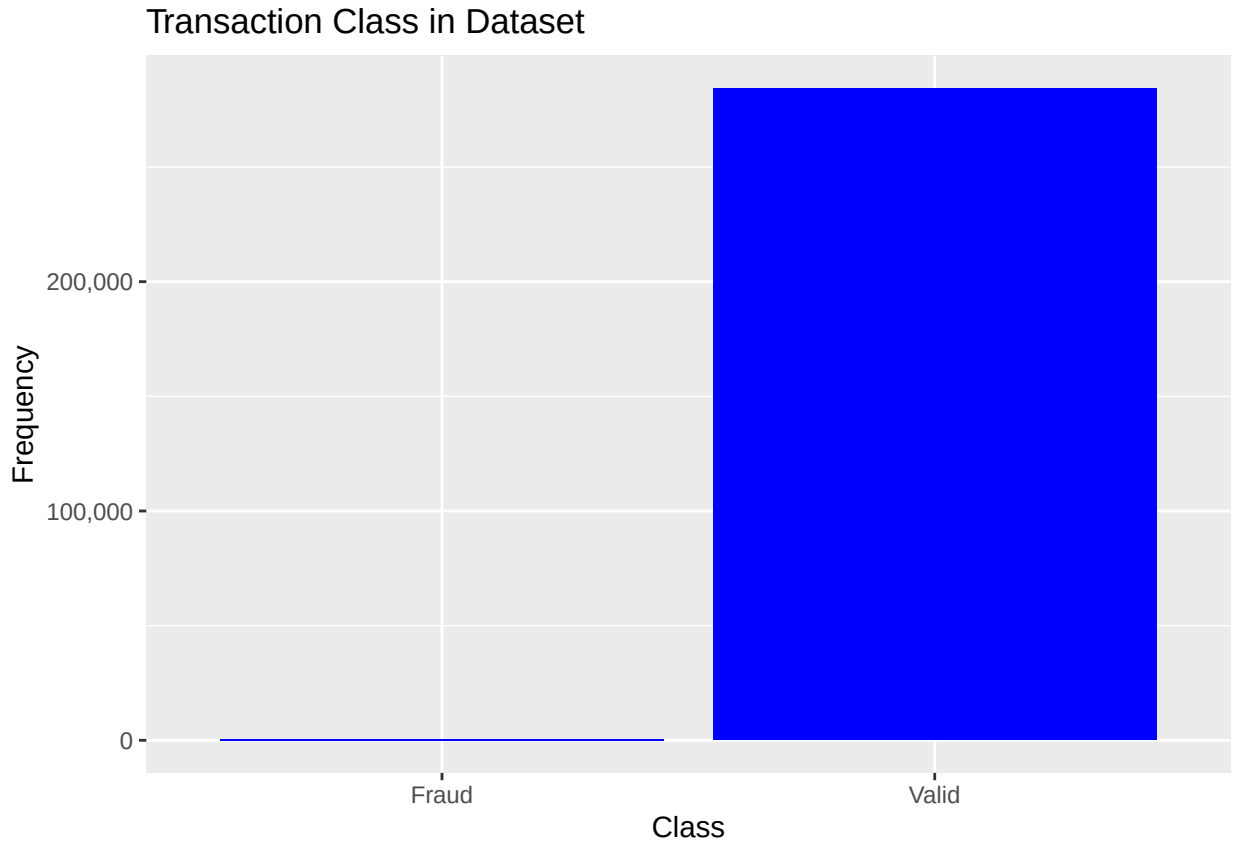
```
## $ V14    <dbl> -0.31116935, -0.14377230, -0.16594592, -0.28792375, -1.11966983~
## $ V15    <dbl> 1.468176972, 0.635558093, 2.345864949, -0.631418118, 0.17512113~
## $ V16    <dbl> -0.47040053, 0.46391704, -2.89008319, -1.05964725, -0.45144918,~
## $ V17    <dbl> 0.207971242, -0.114804663, 1.109969379, -0.684092786, -0.237033~
## $ V18    <dbl> 0.02579058, -0.18336127, -0.12135931, 1.96577500, -0.03819479, ~
## $ V19    <dbl> 0.40399296, -0.14578304, -2.26185710, -1.23262197, 0.80348692, ~
## $ V20    <dbl> 0.25141210, -0.06908314, 0.52497973, -0.20803778, 0.40854236, 0~
## $ V21    <dbl> -0.018306778, -0.225775248, 0.247998153, -0.108300452, -0.00943~
## $ V22    <dbl> 0.277837576, -0.638671953, 0.771679402, 0.005273597, 0.79827849~
## $ V23    <dbl> -0.110473910, 0.101288021, 0.909412262, -0.190320519, -0.137458~
## $ V24    <dbl> 0.06692807, -0.33984648, -0.68928096, -1.17557533, 0.14126698, ~
## $ V25    <dbl> 0.12853936, 0.16717040, -0.32764183, 0.64737603, -0.20600959, -~
## $ V26    <dbl> -0.18911484, 0.12589453, -0.13909657, -0.22192884, 0.50229222, ~
## $ V27    <dbl> 0.133558377, -0.008983099, -0.055352794, 0.062722849, 0.2194222~
## $ V28    <dbl> -0.021053053, 0.014724169, -0.059751841, 0.061457629, 0.2151531~
## $ Amount <dbl> 149.62, 2.69, 378.66, 123.50, 69.99, 3.67, 4.99, 40.80, 93.20, ~
## $ Class  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
```

A single table with an extremely small font may also be used to display all 15 of the entries.

We can view the dimensions of the entire dataset in a table.

We are interested in knowing the ratio of legitimate versus fraudulent transactions. A legitimate transaction is specified as 0, and a fraudulent transaction is defined as 1.

We create a bar graph of the frequency of fraudulent versus legitimate credit card transactions so that the data may be seen.



It is evident that 99.828% of the transactions are legitimate.

Additionally, we may verify that our data set has no missing values.

```
## [1] FALSE
```

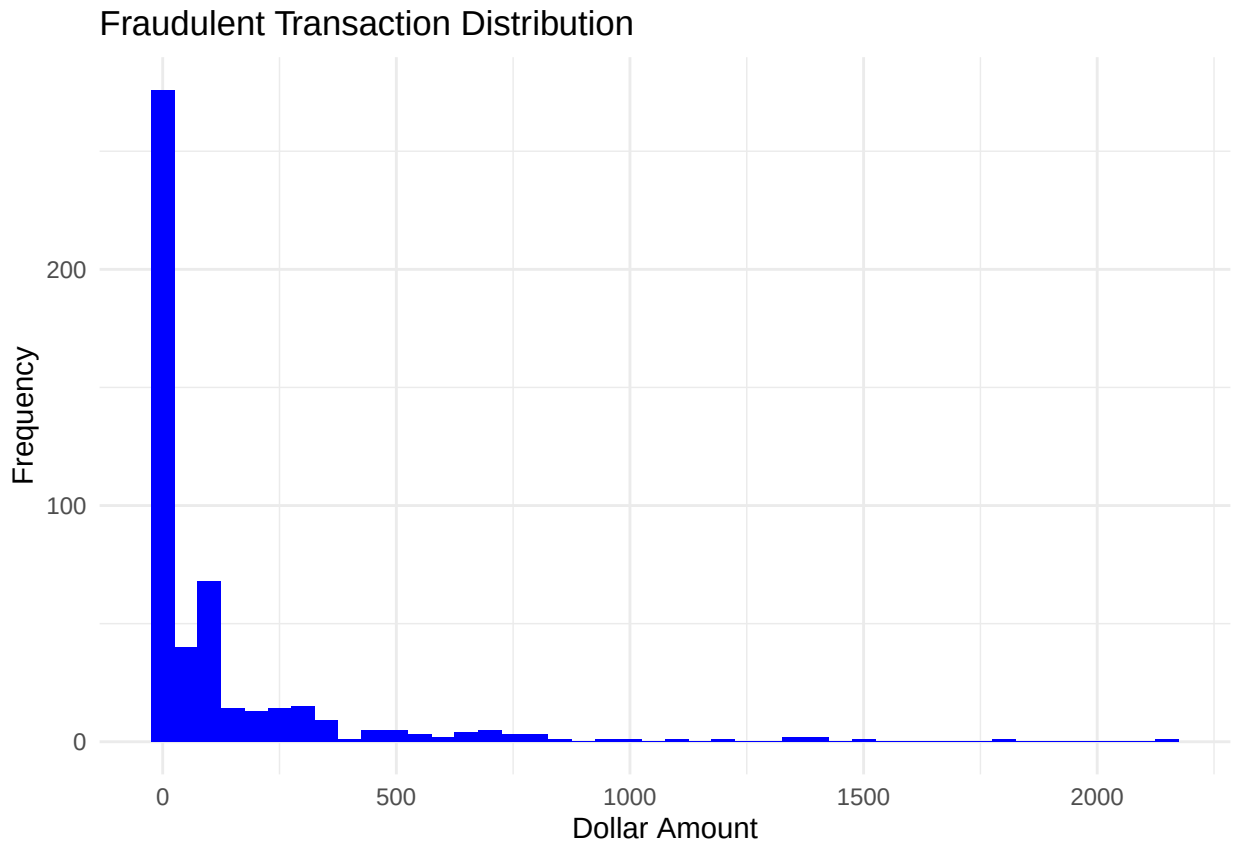
Additionaly, we present a full summary of each variable in the dataset:

```
##       Time              V1                  V2                  V3
##  Min.   :     0   Min.   :-56.40751   Min.   :-72.71573   Min.   :-48.3256
##  1st Qu.: 54202   1st Qu.: -0.92037   1st Qu.: -0.59855   1st Qu.: -0.8904
##  Median : 84692   Median :  0.01811   Median :  0.06549   Median :  0.1799
##  Mean   : 94814   Mean   :  0.00000   Mean   :  0.00000   Mean   :  0.0000
##  3rd Qu.:139320   3rd Qu.:  1.31564   3rd Qu.:  0.80372   3rd Qu.:  1.0272
##  Max.   :172792   Max.   :  2.45493   Max.   : 22.05773   Max.   :  9.3826
##        V4                 V5                  V6                  V7
##  Min.   :-5.68317   Min.   :-113.74331   Min.   :-26.1605   Min.   :-43.5572
##  1st Qu.:-0.84864   1st Qu.:  -0.69160   1st Qu.: -0.7683   1st Qu.: -0.5541
##  Median :-0.01985   Median :  -0.05434   Median : -0.2742   Median :  0.0401
##  Mean   : 0.00000   Mean   :   0.00000   Mean   :  0.0000   Mean   :  0.0000
##  3rd Qu.: 0.74334   3rd Qu.:   0.61193   3rd Qu.:  0.3986   3rd Qu.:  0.5704
##  Max.   :16.87534   Max.   :  34.80167   Max.   : 73.3016   Max.   :120.5895
##        V8                  V9                 V10                 V11
##  Min.   :-73.21672   Min.   :-13.43407   Min.   :-24.58826   Min.   :-4.79747
##  1st Qu.: -0.20863   1st Qu.: -0.64310   1st Qu.: -0.53543   1st Qu.:-0.76249
##  Median :  0.02236   Median : -0.05143   Median : -0.09292   Median :-0.03276
##  Mean   :  0.00000   Mean   :  0.00000   Mean   :  0.00000   Mean   : 0.00000
##  3rd Qu.:  0.32735   3rd Qu.:  0.59714   3rd Qu.:  0.45392   3rd Qu.: 0.73959
```

```
##  Max.   : 20.00721   Max.   : 15.59500   Max.   : 23.74514   Max.   :12.01891
##       V12                 V13                 V14                 V15
##  Min.   :-18.6837   Min.   :-5.79188   Min.   :-19.2143   Min.   :-4.49894
##  1st Qu.: -0.4056   1st Qu.:-0.64854   1st Qu.: -0.4256   1st Qu.:-0.58288
##  Median :  0.1400   Median :-0.01357   Median :  0.0506   Median : 0.04807
##  Mean   :  0.0000   Mean   : 0.00000   Mean   :  0.0000   Mean   : 0.00000
##  3rd Qu.:  0.6182   3rd Qu.: 0.66251   3rd Qu.:  0.4931   3rd Qu.: 0.64882
##  Max.   :  7.8484   Max.   : 7.12688   Max.   : 10.5268   Max.   : 8.87774
##       V16                 V17                 V18
##  Min.   :-14.12985   Min.   :-25.16280   Min.   :-9.498746
##  1st Qu.: -0.46804   1st Qu.: -0.48375   1st Qu.:-0.498850
##  Median :  0.06641   Median : -0.06568   Median :-0.003636
##  Mean   :  0.00000   Mean   :  0.00000   Mean   : 0.000000
##  3rd Qu.:  0.52330   3rd Qu.:  0.39968   3rd Qu.: 0.500807
##  Max.   : 17.31511   Max.   :  9.25353   Max.   : 5.041069
##       V19                 V20                 V21
##  Min.   :-7.213527   Min.   :-54.49772   Min.   :-34.83038
##  1st Qu.:-0.456299   1st Qu.: -0.21172   1st Qu.: -0.22839
##  Median : 0.003735   Median : -0.06248   Median : -0.02945
##  Mean   : 0.000000   Mean   :  0.00000   Mean   :  0.00000
##  3rd Qu.: 0.458949   3rd Qu.:  0.13304   3rd Qu.:  0.18638
##  Max.   : 5.591971   Max.   : 39.42090   Max.   : 27.20284
##       V22                 V23                 V24
##  Min.   :-10.933144   Min.   :-44.80774   Min.   :-2.83663
##  1st Qu.: -0.542350   1st Qu.: -0.16185   1st Qu.:-0.35459
##  Median :  0.006782   Median : -0.01119   Median : 0.04098
##  Mean   :  0.000000   Mean   :  0.00000   Mean   : 0.00000
##  3rd Qu.:  0.528554   3rd Qu.:  0.14764   3rd Qu.: 0.43953
##  Max.   : 10.503090   Max.   : 22.52841   Max.   : 4.58455
##       V25                 V26                 V27
##  Min.   :-10.29540   Min.   :-2.60455   Min.   :-22.565679
##  1st Qu.: -0.31715   1st Qu.:-0.32698   1st Qu.: -0.070840
##  Median :  0.01659   Median :-0.05214   Median :  0.001342
##  Mean   :  0.00000   Mean   : 0.00000   Mean   :  0.000000
##  3rd Qu.:  0.35072   3rd Qu.: 0.24095   3rd Qu.:  0.091045
##  Max.   :  7.51959   Max.   : 3.51735   Max.   : 31.612198
##       V28                 Amount              Class
##  Min.   :-15.43008   Min.   :    0.00   Min.   :0.000000
##  1st Qu.: -0.05296   1st Qu.:    5.60   1st Qu.:0.000000
##  Median :  0.01124   Median :   22.00   Median :0.000000
##  Mean   :  0.00000   Mean   :   88.35   Mean   :0.001728
##  3rd Qu.:  0.07828   3rd Qu.:   77.17   3rd Qu.:0.000000
##  Max.   : 33.84781   Max.   :25691.16   Max.   :1.000000
```

We want to look into the fraud's money amounts. Here, we chart every fraudulent transaction based on its value. There is a significant bias in this plot towards transactions under $100.

Amount	count
1.00	113
0.00	27
99.99	27
0.76	17
0.77	10
0.01	5
2.00	4
3.79	4
0.68	3
1.10	3



We create a table of the ten most frequent fraudulent transactions in order to look into this further. The most fraudulent transaction is by far $1. It's also noteworthy that, in terms of the most frequent fraudulent transactions, a transaction for $0 and a transaction for $99.99 are tied for second place.

We may also look at which valid transactions are most frequently found in the dataset.

One noteworthy finding is that the most frequent fraudulent and legitimate transaction amounts to less than $1. As a matter of fact, a transaction for less than $1 has an approximately five times higher likelihood of being fraudulent than any other transaction in the data set.

A further intriguing finding is that, out of 303 transactions, a transaction worth $99.99 ranks 98th in terms

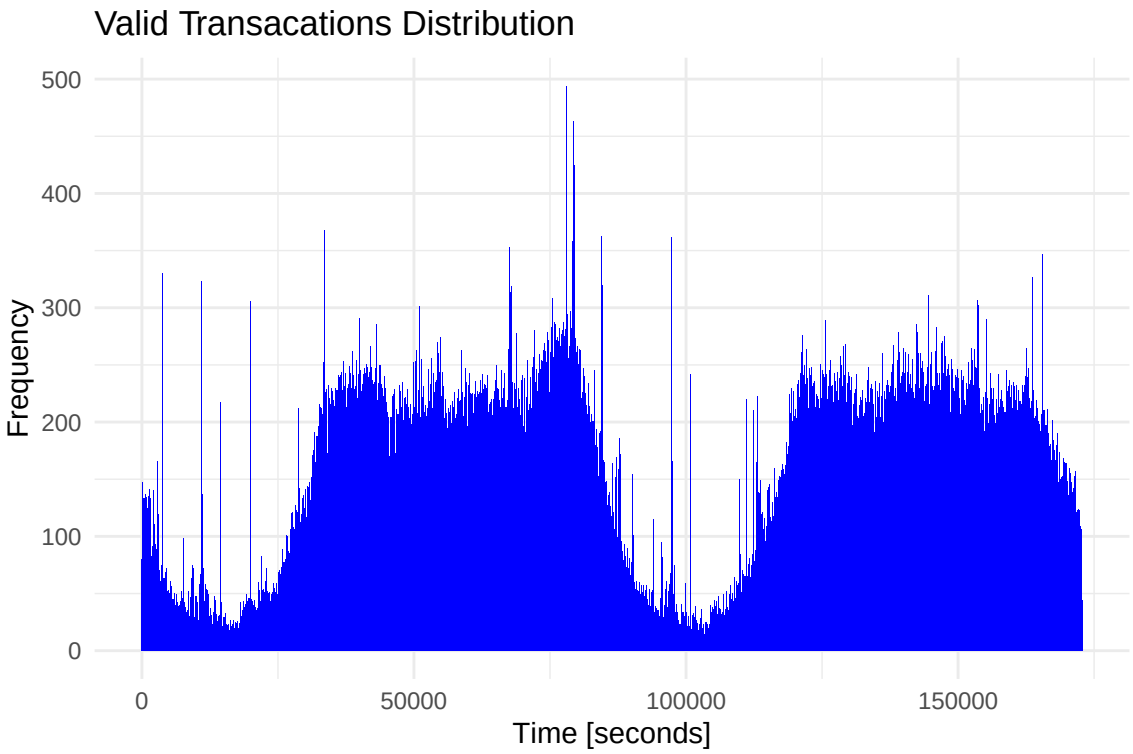
Amount	count
1.00	13575
1.98	6044
0.89	4872
9.99	4746
15.00	3280
0.76	2981
10.00	2950
1.29	2892
1.79	2622
0.99	2304

of validity, but it is tied for second place among fraudulent transactions with 27. This indicates that around 9% of the data set's $99.99 transactions are fraudulent!

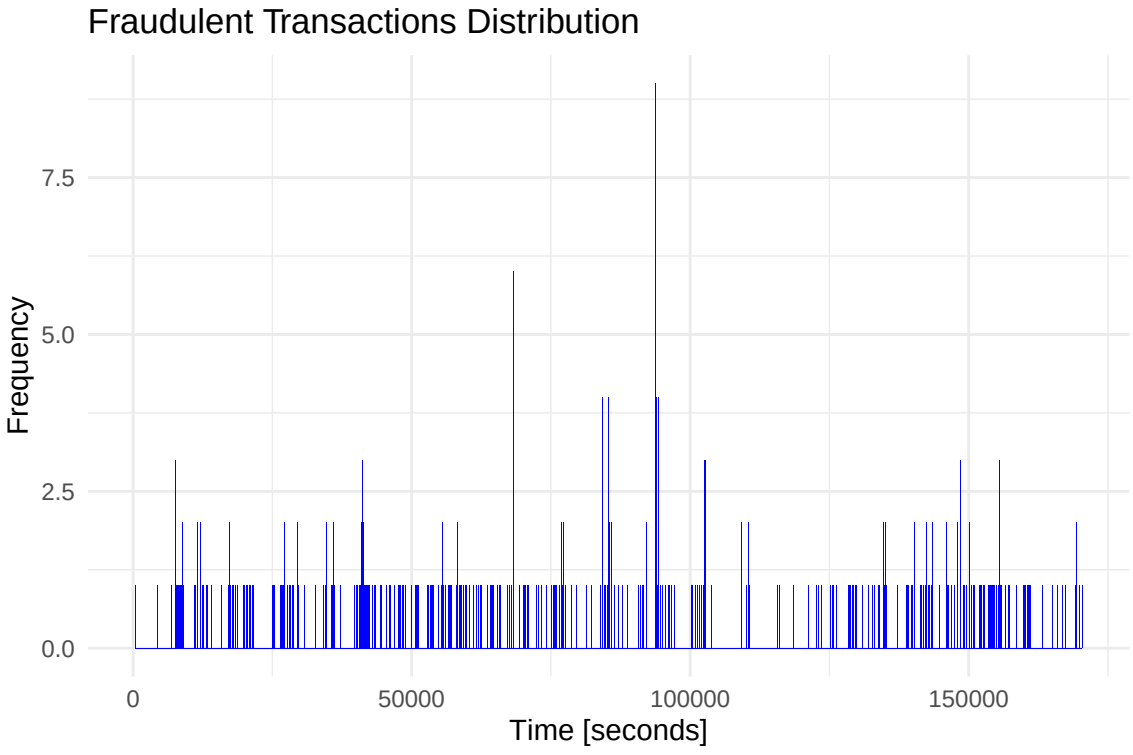
The mean and median transactions for both legitimate and fraudulent transactions are plotted here.

```
## # A tibble: 2 x 3
##   Class `mean(Amount)` `median(Amount)`
##   <int>          <dbl>            <dbl>
## 1     0           88.3            22
## 2     1          122.              9.25
```

A distribution of legitimate transactions over time can be plotted. The episodic distribution of this plot is evident. This makes sense because the approximate duration of this distribution is 86,400 seconds, or one day. The irony is that fewer transactions happen at night while the majority happen during the day. Near the graph's trough, there is a noticeable peak in the number of outlier transactions. We hypothesise that these increases correspond to automated transactions that are completed just before midnight or right after. Bills that are scheduled to automatically paid each month are an example of an automated transaction.



Similarly, to the distribution of valid transactions, we can plot the distribution of fraudulent transactions over time. The fact that there is no obvious episodic distribution suggests that fraud can happen at any time.



Note: We cannot be positive that fraudulent transactions are not episodic without running Fourier analysis (e.g., Fast Fourier Transform) on this data. The frequency distribution depicted above is sufficient to

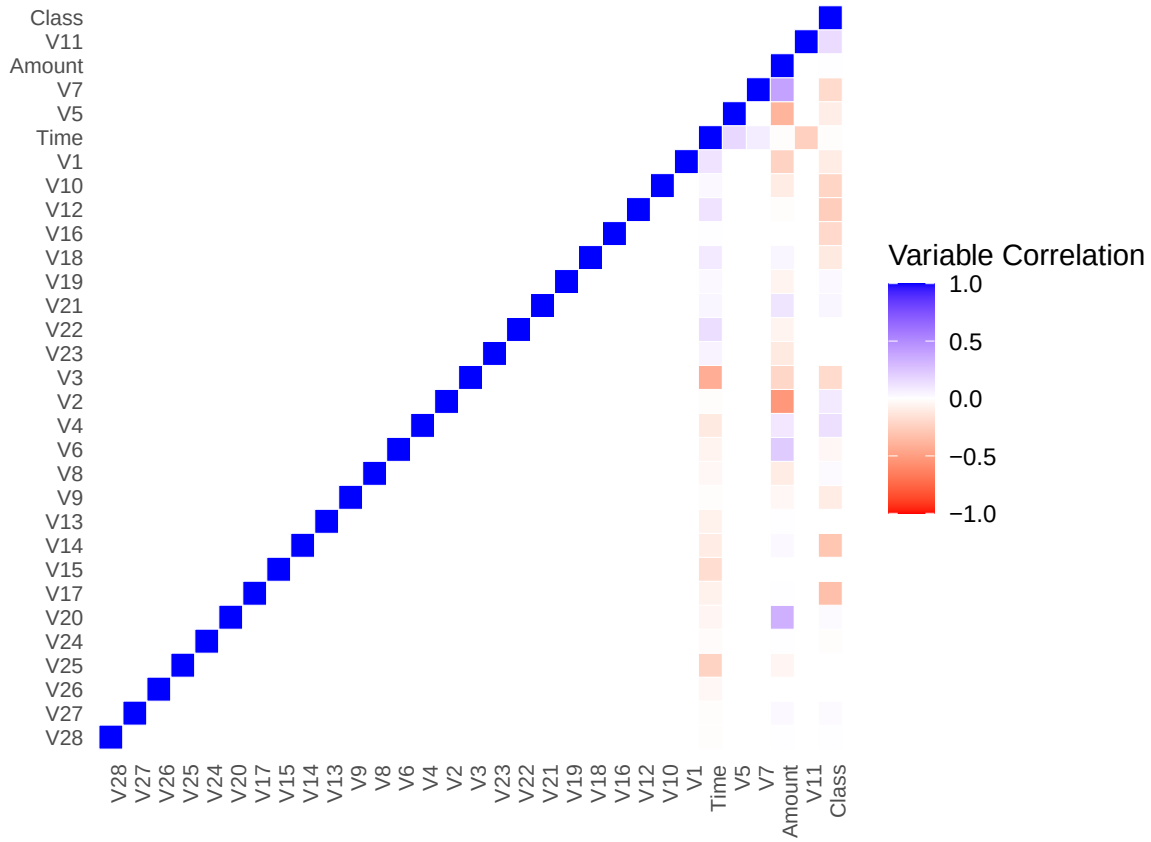
	V28	V27	V26	V25	V24	V20	V17	V15	V14	V13	V9	V8	V6	V4	V2	V3	V23	V22	V21	V19	V18	V16	V12	V10	V1	Time	V5	V7	Amount	V11	Class
V28	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.01	0.00	0.00	0.01	0.00	0.01
V27	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.01	0.00	0.00	0.03	0.00	0.02
V26	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.04	0.00	0.00	0.00	0.00	0.00
V25	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.23	0.00	0.00	-0.05	0.00	0.00
V24	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.02	0.00	0.00	0.01	0.00	-0.01
V20	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.05	0.00	0.00	0.34	0.00	0.02
V17	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.07	0.00	0.00	0.01	0.00	-0.33
V15	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.18	0.00	0.00	0.00	0.00	0.00
V14	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.10	0.00	0.00	0.03	0.00	-0.30
V13	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.07	0.00	0.00	0.01	0.00	0.00
V9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.01	0.00	0.00	-0.04	0.00	-0.10
V8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.04	0.00	0.00	-0.10	0.00	0.02
V6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.06	0.00	0.00	0.22	0.00	-0.04
V4	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.11	0.00	0.00	0.10	0.00	0.13
V2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.01	0.00	0.00	-0.53	0.00	0.09
V3	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.42	0.00	0.00	-0.21	0.00	-0.19
V23	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.00	0.00	-0.11	0.00	0.00
V22	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.14	0.00	0.00	-0.06	0.00	0.00
V21	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.11	0.00	0.04
V19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	-0.06	0.00	0.03
V18	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.09	0.00	0.00	0.04	0.00	-0.11
V16	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	-0.20
V12	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.12	0.00	0.00	-0.01	0.00	-0.26
V10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.03	0.00	0.00	-0.10	0.00	-0.22
V1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.12	0.00	0.00	-0.23	0.00	-0.10
Time	-0.01	-0.01	-0.04	-0.23	-0.02	-0.05	-0.07	-0.18	-0.10	-0.07	-0.01	-0.04	-0.06	-0.11	-0.01	-0.42	0.05	0.14	0.04	0.03	0.09	0.01	0.12	0.03	0.12	1.00	0.17	0.08	-0.01	-0.25	-0.01
V5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.17	1.00	0.00	-0.39	0.00	-0.09
V7	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.00	1.00	0.40	0.00	-0.19
Amount	0.01	0.03	0.00	-0.05	0.01	0.34	0.01	0.00	0.03	0.01	-0.04	-0.10	0.22	0.10	-0.53	-0.21	-0.11	-0.06	0.11	-0.06	0.04	0.00	-0.01	-0.10	-0.23	-0.01	-0.39	0.40	1.00	0.00	0.01
V11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.25	0.00	0.00	0.00	1.00	0.15
Class	0.01	0.02	0.00	0.00	-0.01	0.02	-0.33	0.00	-0.30	0.00	-0.10	0.02	-0.04	0.13	0.09	-0.19	0.00	0.00	0.04	0.03	-0.11	-0.20	-0.26	-0.22	-0.10	-0.01	-0.09	-0.19	0.01	0.15	1.00

demonstrate that fraudulent transactions are not episodic and can happen at any time; this analysis is outside the purview of this research.

We want to graph the variables and determine their association. First, a correlation matrix is created.

This is a matrix showing how the 31 different variables are correlated.

Further, we can plot the correlation. Observe how the correlation coefficients between all of the variables, V1 through V28, are incredibly low, particularly when it comes to the 'Class' feature. Given that PCA was used to process the data, this was already anticipated.



Since fraud does not seem to be related to a particular time of day, the 'Time' variable will no longer be included in the dataset.

Using the head() function, we can see the first six items and see that the variable "Time" has been eliminated.

```
##           V1          V2        V3         V4          V5          V6
## 1 -1.3598071 -0.07278117 2.5363467  1.3781552 -0.33832077  0.46238778
## 2  1.1918571  0.26615071 0.1664801  0.4481541  0.06001765 -0.08236081
## 3 -1.3583541 -1.34016307 1.7732093  0.3797796 -0.50319813  1.80049938
## 4 -0.9662717 -0.18522601 1.7929933 -0.8632913 -0.01030888  1.24720317
## 5 -1.1582331  0.87773675 1.5487178  0.4030339 -0.40719338  0.09592146
## 6 -0.4259659  0.96052304 1.1411093 -0.1682521  0.42098688 -0.02972755
##            V7          V8         V9         V10        V11         V12
## 1  0.23959855  0.09869790  0.3637870  0.09079417 -0.5515995 -0.61780086
## 2 -0.07880298  0.08510165 -0.2554251 -0.16697441  1.6127267  1.06523531
## 3  0.79146096  0.24767579 -1.5146543  0.20764287  0.6245015  0.06608369
## 4  0.23760894  0.37743587 -1.3870241 -0.05495192 -0.2264873  0.17822823
## 5  0.59294075 -0.27053268  0.8177393  0.75307443 -0.8228429  0.53819555
## 6  0.47620095  0.26031433 -0.5686714 -0.37140720  1.3412620  0.35989384
##          V13        V14        V15        V16         V17         V18
## 1 -0.9913898 -0.3111694  1.4681770 -0.4704005  0.20797124  0.02579058
## 2  0.4890950 -0.1437723  0.6355581  0.4639170 -0.11480466 -0.18336127
## 3  0.7172927 -0.1659459  2.3458649 -2.8900832  1.10996938 -0.12135931
## 4  0.5077569 -0.2879237 -0.6314181 -1.0596472 -0.68409279  1.96577500
## 5  1.3458516 -1.1196698  0.1751211 -0.4514492 -0.23703324 -0.03819479
## 6 -0.3580907 -0.1371337  0.5176168  0.4017259 -0.05813282  0.06865315
##           V19         V20          V21          V22         V23         V24
## 1  0.40399296  0.25141210 -0.018306778  0.277837576 -0.11047391  0.06692807
## 2 -0.14578304 -0.06908314 -0.225775248 -0.638671953  0.10128802 -0.33984648
## 3 -2.26185710  0.52497973  0.247998153  0.771679402  0.90941226 -0.68928096
## 4 -1.23262197 -0.20803778 -0.108300452  0.005273597 -0.19032052 -1.17557533
## 5  0.80348692  0.40854236 -0.009430697  0.798278495 -0.13745808  0.14126698
## 6 -0.03319379  0.08496767 -0.208253515 -0.559824796 -0.02639767 -0.37142658
##          V25        V26          V27         V28 Amount Class
## 1  0.1285394 -0.1891148  0.133558377 -0.02105305 149.62     0
## 2  0.1671704  0.1258945 -0.008983099  0.01472417   2.69     0
## 3 -0.3276418 -0.1390966 -0.055352794 -0.05975184 378.66     0
## 4  0.6473760 -0.2219288  0.062722849  0.06145763 123.50     0
## 5 -0.2060096  0.5022922  0.219422230  0.21515315  69.99     0
## 6 -0.2327938  0.1059148  0.253844225  0.08108026   3.67     0
```

III. Methods and Analysis

For this report we will investigate two models: the K-Nearest Neighbor Model, and the Random Forest Model.

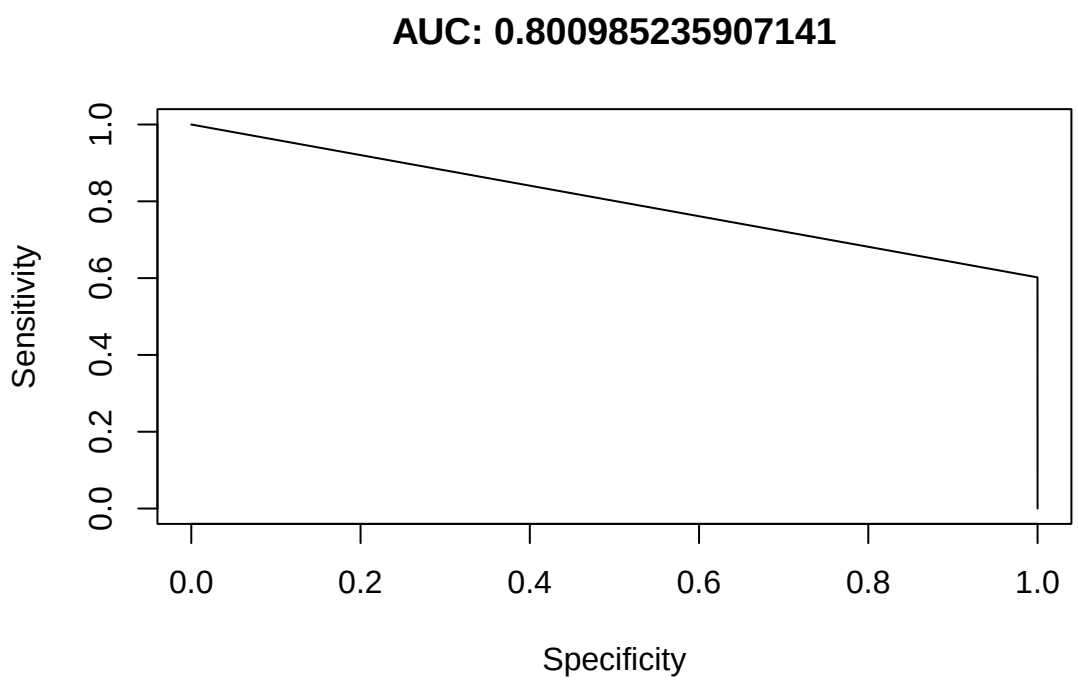
III.A. K-Nearest Neighbor The non-parametric technique known as the K-Nearest Neighbors algorithm (KNN) is utilized for classification, where the input comprises the k closest instances from the training set in the feature space. When applying KNN for classification, which involves determining the validity or fraudulence of a transaction, the output represents class membership. The classification of an object is determined by a majority vote from its neighbors, assigning the object to the class that is most prevalent among its k nearest neighbors. Various k values were experimented with, and 5 was selected as the optimal value yielding the best results. In this model, the target is 'Class,' and all other variables serve as predictors.

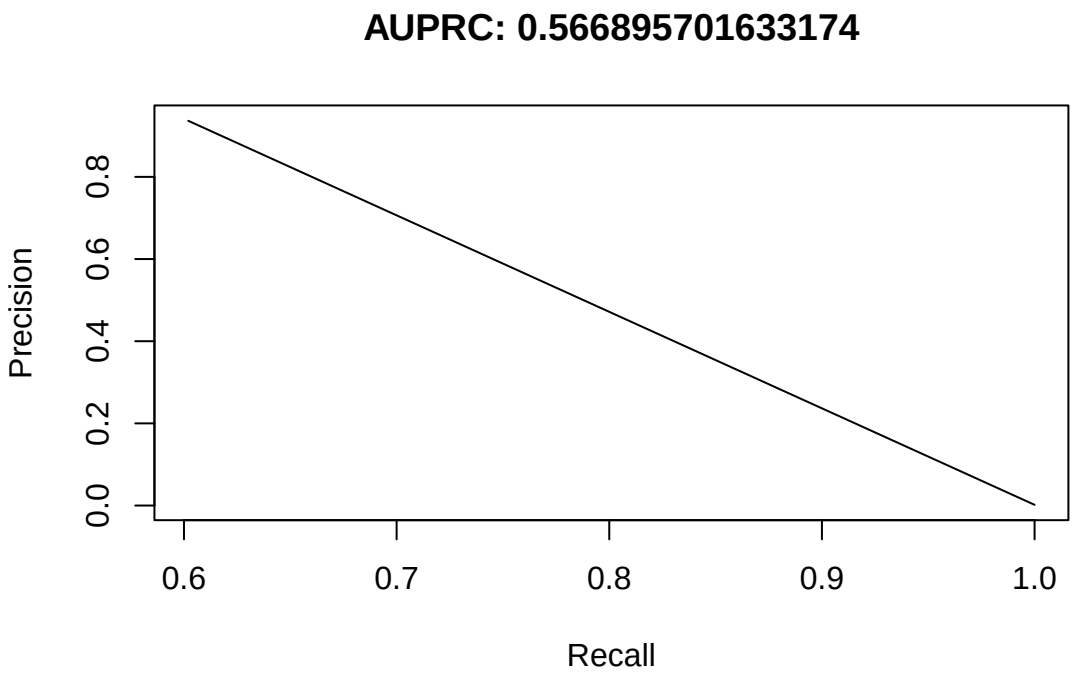
III.B. Random Forest The algorithm known as Random Forest (sometimes referred to as Random Decision Forests) is a machine learning algorithm wherein a classification ensemble learning method is employed. During the training phase, the algorithm constructs numerous decision trees and, during the classification process, determines the class that represents the mode of classification across the individual trees. These decision trees function as a sequence from observations about an item (depicted in the branches) to conclusions regarding the item's target value (depicted in the leaves). In this particular model, the target is 'Class,' which denotes whether a transaction is valid or fraudulent, while all other variables serve as predictors. The specified number of trees for this model is set at 500.

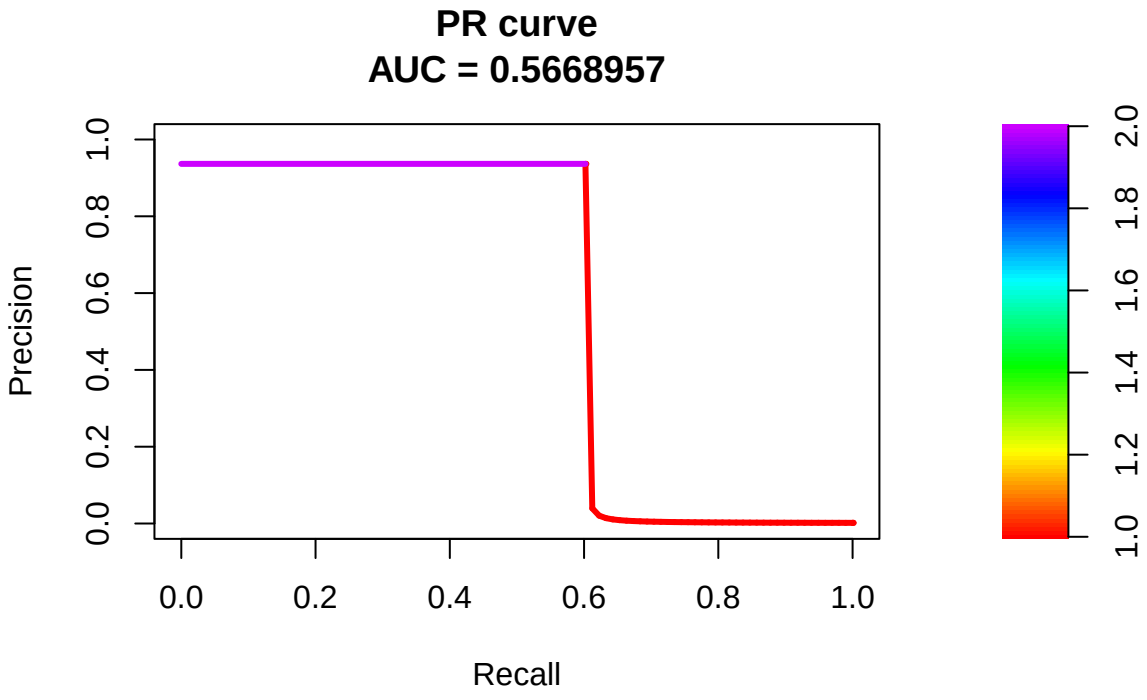
IV. Results

We divide the dataset into three sets before doing any computations: a training set, a test set, and a cross-validation set.

IV.A. K-Nearest Neighbor For the K Nearest Neighbors model, the AUC is about 0.8. However, for the AUPRC, it is a value of 0.57. The goal is of the AUC of 0.8 has been met.





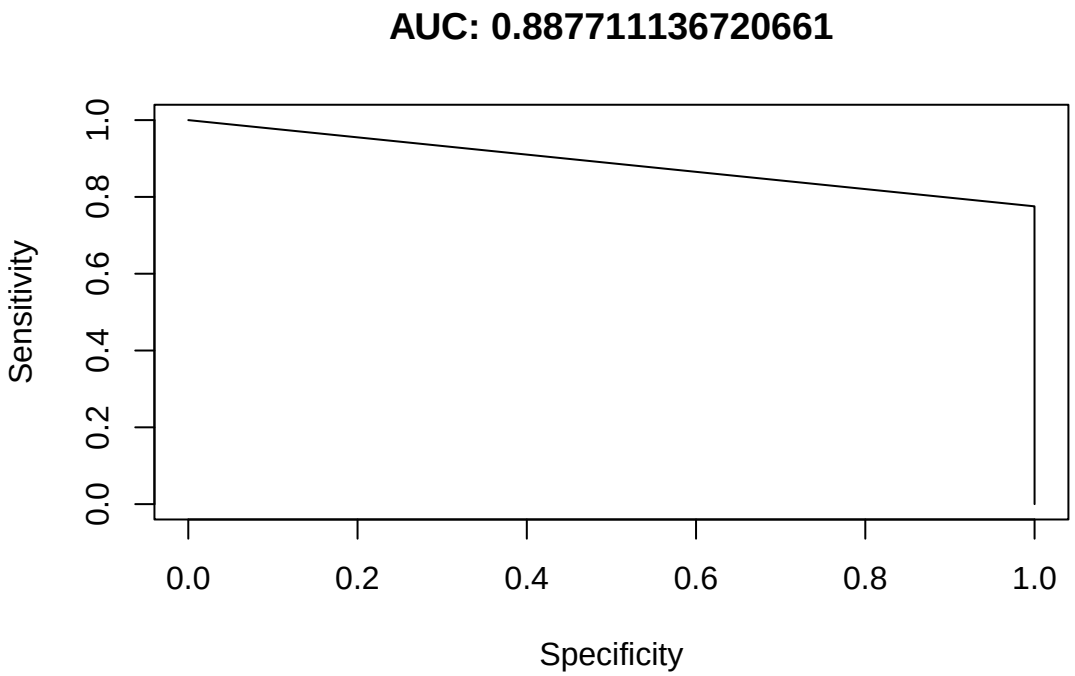


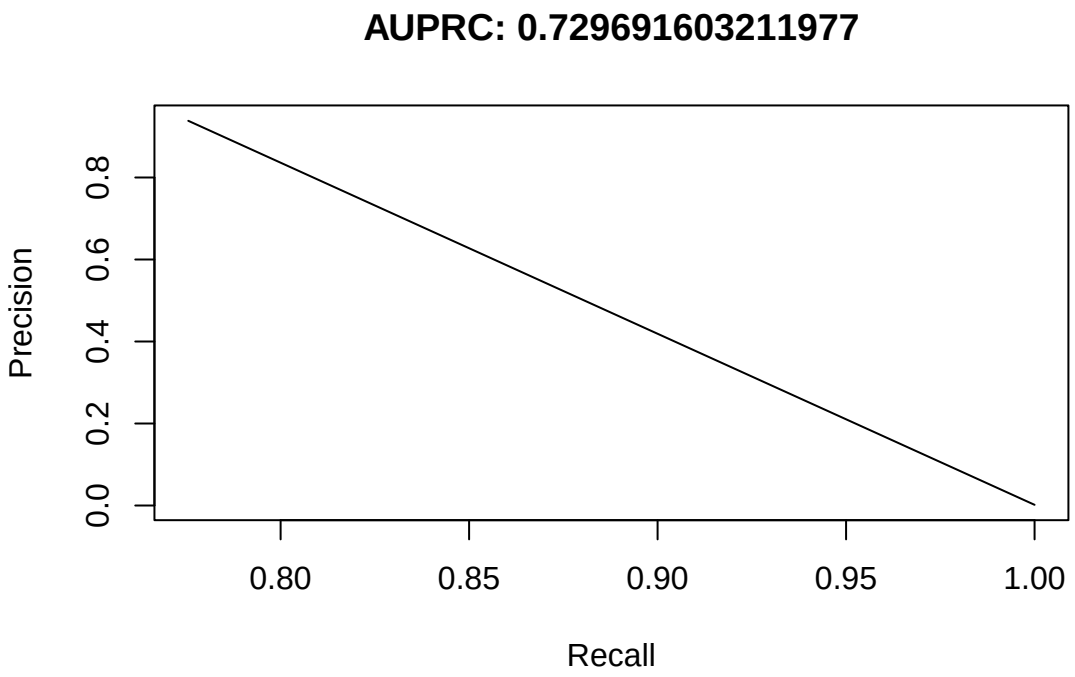
In a data frame, we store and present the outcomes of our K-Nearest Neighbour Model alongside other findings.

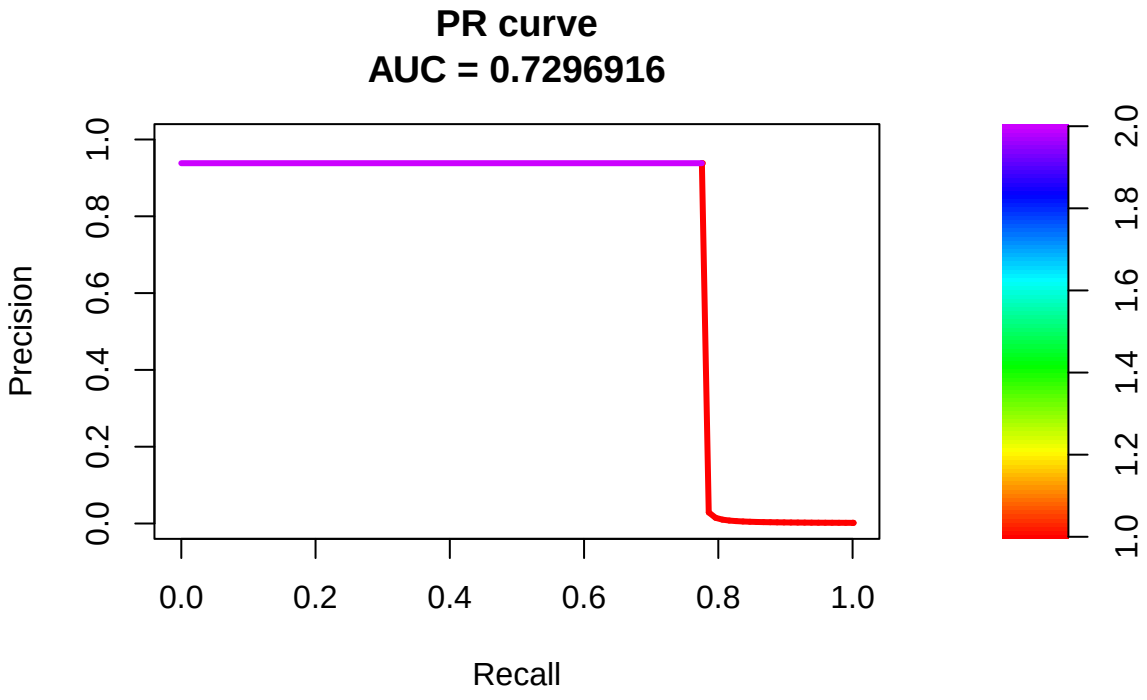
```
##                   Model       AUC     AUPRC
## 1 K-Nearest Neighbors 0.8009852 0.5668957
```

Model	AUC	AUPRC
K-Nearest Neighbors	0.8009852	0.5668957

IV.B. Random Forest In the case of our Random Forest Model, we not only achieve the highest AUC for sensitivity versus specificity (0.88) but also secure the top AUC for precision versus recall (0.8). Among the developed and trained models, this particular model proves to be the most accurate for our intended task. The utilization of 500 trees in this algorithm proves to be effective.







Our Random Forest Model results are saved in a data frame, where they are shown alongside earlier findings.

Model	AUC	AUPRC
K-Nearest Neighbors	0.8009852	0.5668957
Random Forest	0.8877111	0.7296916

V. Conclusion

In this report, we use a machine learning strategy to handle credit card fraud. We are presented with a machine learning task that makes use of the model's accuracy by calculating the Area Under the Precision-Recall Curve rather than a more conventional way like a confusion matrix because credit card theft is extremely rare in comparison to the volume of valid transactions.

A Kaggle-provided dataset of credit card transactions was used to evaluate the two generated models. The results from the two models that were used to create this report are once more shown below.

```
##                 Model       AUC     AUPRC
## 1 K-Nearest Neighbors 0.8009852 0.5668957
## 2       Random Forest 0.8877111 0.7296916
```

The Random Forest approach was the model that most closely fit the requirements of the given task. This machine learning algorithm is a classification technique that uses ensemble learning. During training, it builds a large number of decision trees and outputs the class that represents the average categorization of each individual tree. We choose 500 as the maximum number of trees in our approach.

When compared with one other models that was previously evaluated on this dataset, our Random Forest method findings are striking. We calculated the Area Under the Precision-Recall Curve (AUPRC) to be 0.73 and the Area Under the Curve (AUC) for sensitivity vs specificity to be 0.887. This model significantly increased the K-Nearest Neighbours algorithm's AUPRC. Higher-level models in machine learning might be able to produce superior outcomes. These models, however, are outside the purview of the project and this course/ project.