Coursework: EDA & Regression

Statistics and Machine Learning 1 - MSc Data Science

11549067

1. Brief description of the Data

This dataset contains information on rail journeys made by passengers in the United Kingdom between 1 January and 30 April 2024. In total, the dataset includes records of 31,645 journeys with 18 attributes like the payment method, departure and arrival stations, type of tickets, prices, and dates and hours of departure and arrival.

2. Exploratory Data Analysis

2.1. Journey Details and Insights

Out of all recorded journeys, 87% (27,479) were on time, while 7% (2,289) were delayed, and 6% (1,877) were cancelled. We do not observe significant amounts of missing data. The 1,880 missing values in *Actual.Arrival* correspond to the 1,877 cancelled journeys that never reached their destination, plus 3 missing departure stations:

```
##   Payment.Method Railcard Ticket.Class Ticket.Type Price Departure.Station
## 1              0        0            0           0     0                  0
##   Arrival.Station Departure Scheduled.Arrival Actual.Arrival Journey.Status
## 1               0         3                 4           1880              0
##   Reason.for.Delay Refund.Request
## 1                0              0
```

The distribution of journeys is relatively consistent across the four months of data collection, with an average of 7,440 journeys per month.

```
##   Month   Nº        %
## 1   Jan 8107 25.62101
## 2   Feb 7642 24.15144
## 3   Mar 8113 25.63997
## 4   Apr 7780 24.58757
```

One inconsistency was observed: in 914 journeys, the train arrived at its destination *before* it departed from the station:

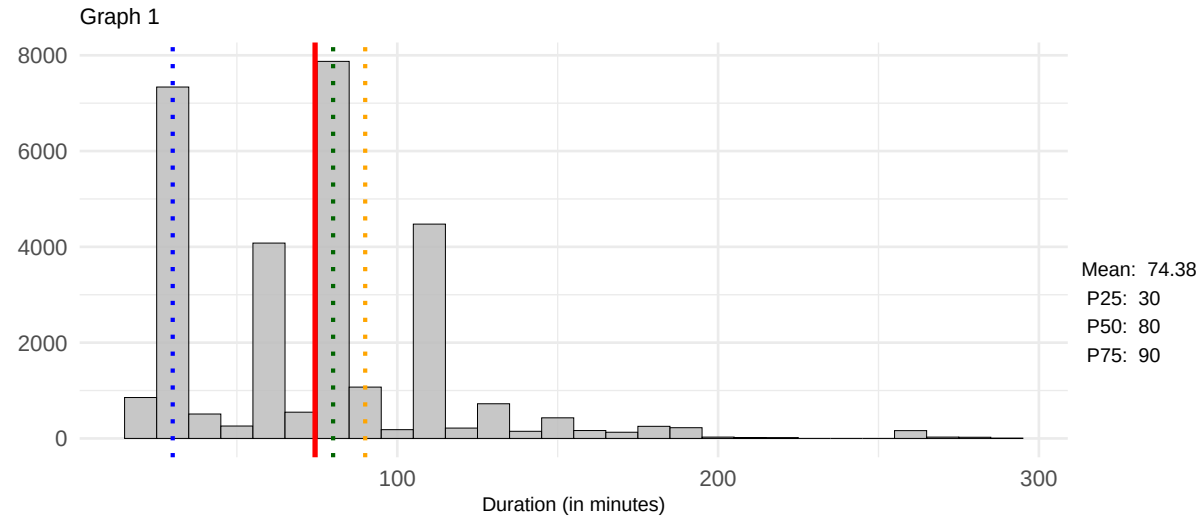
```
## [1] 914
```

This issue likely occurs because the dataset did not account for journeys that span across midnight. Here is an example:

```
##      Departure.Station   Arrival.Station           Departure
## 1702           Reading London Paddington 2024-01-09 23:45:00
##           Actual.Arrival
## 1702 2024-01-09 00:10:00
```

To resolve it, we identified journeys where the arrival time was earlier than the departure time, and one day to the *Actual.Arrival* timestamp was added. With the adjusted distribution of journey duration, we found an average duration of 74 minutes. Half of the passengers experienced a journey duration of at least one and a half hours.

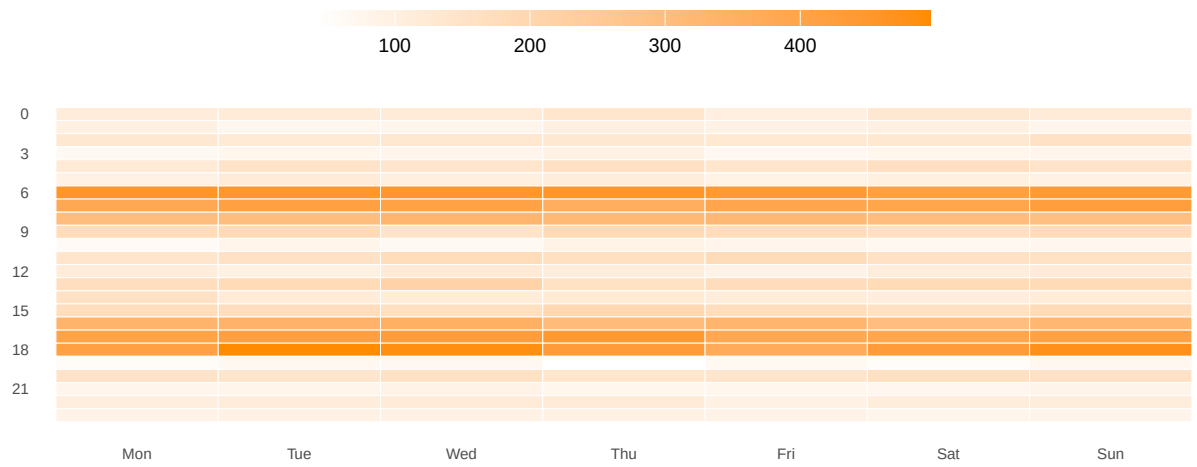
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   15.00   30.00   80.00   74.38   90.00  288.00    1882
```



Ten station combinations accounted for 80% of all journeys. Graph 2 shows the most common departure and arrival stations and the connections between them.

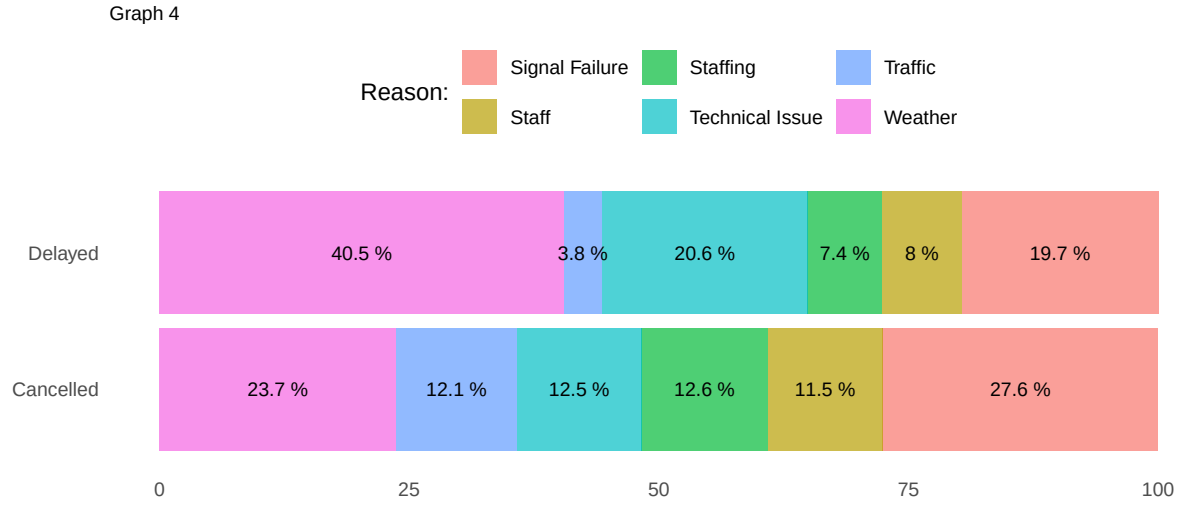
The data also highlights clear peak travel hours. As shown in Graph 3, there is a marked increase in the number of journeys between 6:00 am and 9:00 am, and between 4:00 pm and 7:00 pm.

Graph 3



Graph 4 displays the main reasons for delayed and cancelled journeys. The most common causes are bad weather, signal failures, and technical issues.

Graph 4



2.2. Tickets and Likelihood of Refund Requests

The average ticket price was £23, with a higher concentration of tickets at the lower end of the price range. 25% of tickets cost £5 or less and at least half of the tickets were priced at £11 or less.

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    5.00   11.00   23.43   35.00  267.00
```

We observe that 1,114 (27%) passengers requested a refund, all from delayed or cancelled journeys. Refunds were more common for cancelled journeys than for delayed ones.

Table 1: Refund Request vs Journey Status

	No	Yes
Cancelled	0.6963239	0.3036761
Delayed	0.7623416	0.2376584
On Time	1.0000000	0.0000000

The proportion of passengers requesting a refund is roughly the same across the price range, except when the ticket price exceeds £35, with a slight increase to 7%:

Table 2: Refund Request vs Price

	No	Yes
Less than 6	0.9747842	0.0252158
Between 6 and 11	0.9678183	0.0321817
Between 12 and 23	0.9757824	0.0242176
Between 24 and 35	0.9687861	0.0312139
More than 35	0.9317248	0.0682752

A difference in the proportion of refund requests is also observed depending on the type of railcard used: 9% of passengers who used an "Adult" railcard requested a refund:

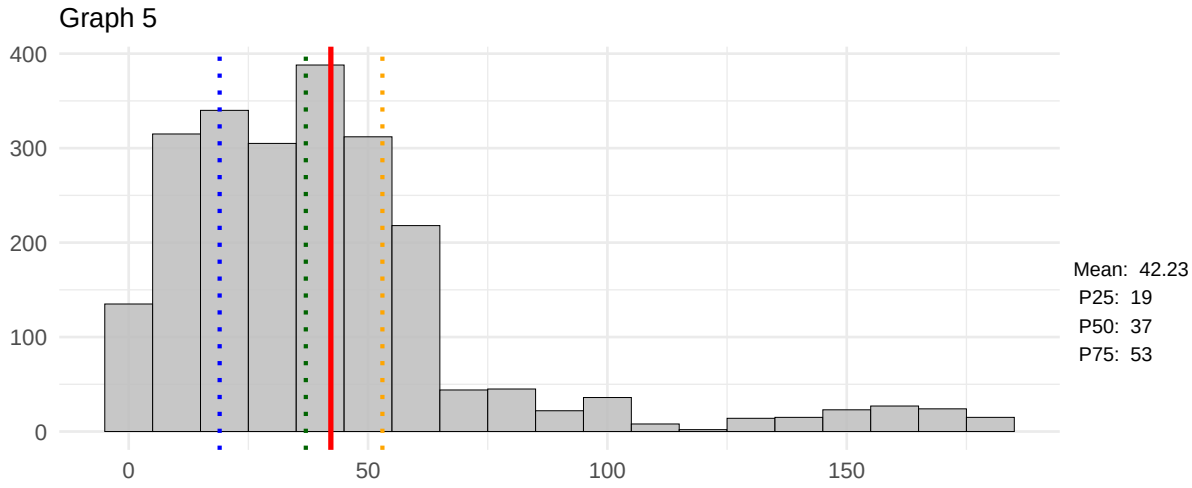
Table 3: Refund Request vs Railcard

	No	Yes
Adult	0.9128999	0.0871001
Disabled	0.9805762	0.0194238
None	0.9755153	0.0244847
Senior	0.9571429	0.0428571

3. Column *DelayInMinutes*

We created a new variable *DelayInMinutes*, which calculates the delay duration for all delayed journeys. The average delay duration is 42 minutes. Of the 2,289 passengers with delayed journeys, half of them experienced a delay of 37 minutes or more, while 75% faced a delay of at least 19 minutes.

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00   19.00   37.00   42.23   53.00  180.00   29357
```



4. Modeling probability of refund

Now, we will focus on the journeys that were either delayed or cancelled. We will fit an appropriate regression model to predict the probability that a passenger requests a refund. First, a variable named *MediumPrice* will be created for tickets priced above £10 and up to £30:

```
dataModel = MavenRail %>%
  filter(Journey.Status != "On Time") %>% #Journey.Status is not On Time
  mutate(MediumPrice = ifelse(Price > 10 & Price <= 30, 1, 0))
```

```
##
##    0    1
## 3417  749
```

We fit a Generalized Linear Model (GLM) from the binomial family, as the response variable is categorical with two levels, using *MediumPrice* as the sole predictor. The model can be formally expressed like this:

$$\log\left[\frac{P(\text{Refund. Request}=1)}{1-P(\text{Refund. Request}=1)}\right] = \alpha + \beta_1(\text{MediumPrice}) \tag{1}$$

MediumPrice shows a positive and statistically significant effect on the probability of requesting a refund.

4.1. Probability given that they paid £5

A £5 ticket price lies outside the range of *MediumPrice*, so the value of this variable in the formula is equal to zero. The right side of the equation, thus, simplifies to -1.076, giving us the odds expressed as follows:

$$\frac{\text{P(Refund.Request)}}{1-\text{P(Refund.Request)}} = e^{-1.076} = 0.341$$

Solving the equation shows that the probability of requesting a refund when the ticket price is £5 is 25%.

$$P(\text{Refund.Request}) = \frac{0.341}{1.341} = 0.254$$

```
## [1] 0.2543168
```

Table 4: Binomial regression with one predictor

	Model 1
(Intercept)	−1.076***
	[−1.153, −0.999]
MediumPrice	0.354***
	[0.182, 0.524]
Num.Obs.	4166
AIC	4826.0
BIC	4838.6
Log.Lik.	−2410.983
F	16.507
RMSE	0.44

+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001

4.2. Probability given that they paid £25

A ticket price of £25 falls within the range of *MediumPrice*, so the value on the right side of the formula is −0.722. This results in the odds being expressed as follows:

$$\frac{\text{P(Refund.Request)}}{1-\text{P(Refund.Request)}} = e^{-0.722} = 0.486$$

Solving the equation shows that the probability of requesting a refund when the ticket price is £25 is 33%.

$$P(\text{Refund.Request}) = \frac{0.486}{1.486} = 0.327$$

```
## [1] 0.3271024
```

5. Model Fit and Prediction

5.1. Training Models

Using *MavenRail*, we will fit a regression model to predict the likelihood of passengers requesting a refund. Since no refund requests occur for on-time journeys in *MavenRail*, it could be assumed that passengers in this category will not request a refund. However, the model should identify that the probability in those scenarios should be zero. Thus, we trained models using both the full dataset and a subset of delayed and cancelled journeys.

Table 5: Regression Models and Predictors

Group	**Model**	**Predictors**
Whole dataset	1	Price
Whole dataset	2	Price, Railcard
Whole dataset	3	Price, Railcard, Ticket class, Ticket type
Whole dataset	4	Price, Railcard, Ticket class, Ticket type, Journey status
Only ”delayed” or ”cancelled” journeys	5	Price
Only ”delayed” or ”cancelled” journeys	6	Price, Railcard
Only ”delayed” or ”cancelled” journeys	7	Price, Railcard, Ticket class, Ticket type
Only ”delayed” or ”cancelled” journeys	8	Price, Railcard, Ticket class, Ticket type, Journey status
Only ”delayed” or ”cancelled” journeys	9	Price, Railcard, Ticket class, Ticket type, Journey status, Reason for delay

Data was divided into training and testing sets. To address the imbalance in the response variable, stratified sampling was applied to maintain consistent proportions across both sets.

Table 6: Modeling probability of requesting refund (whole dataset)

	Model 1	Model 2	Model 3	Model 4
(Intercept)	−3.679***	−2.826***	−3.349***	−0.300
	[−3.834, −3.532]	[−3.001, −2.658]	[−3.646, −3.061]	[−0.674, 0.069]
PriceNEWBetween 6 and 11	0.293**	0.423***	0.503***	0.561***
	[0.083, 0.503]	[0.210, 0.636]	[0.288, 0.719]	[0.293, 0.830]
PriceNEWBetween 12 and 23	−0.004	0.314**	0.400**	0.726***
	[−0.240, 0.228]	[0.074, 0.551]	[0.158, 0.640]	[0.424, 1.028]
PriceNEWBetween 24 and 35	0.279*	0.710***	0.888***	1.437***
	[0.017, 0.533]	[0.441, 0.973]	[0.615, 1.155]	[1.090, 1.784]
PriceNEWMore than 35	1.052***	1.339***	1.586***	1.231***
	[0.865, 1.243]	[1.143, 1.538]	[1.376, 1.799]	[0.965, 1.503]
RailcardDisabled		−1.577***	−1.535***	−1.872***
		[−1.894, −1.282]	[−1.853, −1.240]	[−2.253, −1.508]
RailcardNone		−1.521***	−1.557***	−1.975***
		[−1.674, −1.367]	[−1.711, −1.403]	[−2.200, −1.753]
RailcardSenior		−0.669***	−0.600***	−0.648***
		[−0.919, −0.429]	[−0.851, −0.358]	[−0.988, −0.311]
Ticket.ClassStandard			0.544***	0.400**
			[0.314, 0.785]	[0.115, 0.693]
Ticket.TypeAnytime			−0.667***	−0.864***
			[−0.890, −0.452]	[−1.120, −0.616]
Ticket.TypeOff-Peak			0.091	0.364***
			[−0.063, 0.243]	[0.174, 0.553]
Journey.StatusDelayed				−0.837***
				[−1.033, −0.643]
Journey.StatusOn Time				−22.007
				[−271.989, −251.902]
Num.Obs.	25 317	25 317	25 317	25 317
AIC	7570.3	7204.1	7138.9	3413.3
BIC	7611.0	7269.2	7228.4	3519.1
Log.Lik.	−3780.160	−3594.045	−3558.439	−1693.642
F	43.353	80.943	64.136	32.586
RMSE	0.18	0.18	0.18	0.15

+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001

Table 7: Modeling probability of requesting refund (cancelled and delayed)

	Model 5	Model 6	Model 7	Model 8	Model 9
(Intercept)	−1.177***	−0.444***	−0.882***	−0.382*	−0.992***
	[−1.344, −1.016]	[−0.648, −0.242]	[−1.235, −0.535]	[−0.755, −0.013]	[−1.408, −0.581]
PriceNEWBetween 6 and 11	0.382**	0.474***	0.505***	0.506***	0.444**
	[0.143, 0.621]	[0.216, 0.733]	[0.243, 0.768]	[0.241, 0.773]	[0.167, 0.722]
PriceNEWBetween 12 and 23	0.203	0.760***	0.847***	0.726***	0.499**
	[−0.059, 0.464]	[0.475, 1.046]	[0.554, 1.140]	[0.429, 1.024]	[0.194, 0.803]
PriceNEWBetween 24 and 35	0.433**	1.152***	1.334***	1.329***	1.023***
	[0.124, 0.737]	[0.816, 1.485]	[0.990, 1.677]	[0.979, 1.678]	[0.662, 1.382]
PriceNEWMore than 35	0.086	0.623***	0.819***	1.127***	1.013***
	[−0.122, 0.297]	[0.389, 0.861]	[0.572, 1.070]	[0.864, 1.396]	[0.736, 1.294]
RailcardDisabled		−1.504***	−1.528***	−1.858***	−1.520***
		[−1.857, −1.165]	[−1.884, −1.187]	[−2.232, −1.500]	[−1.918, −1.135]
RailcardNone		−1.655***	−1.684***	−1.985***	−1.533***
		[−1.855, −1.456]	[−1.890, −1.481]	[−2.210, −1.765]	[−1.782, −1.287]
RailcardSenior		−0.131	−0.154	−0.546**	−0.119
		[−0.449, 0.184]	[−0.475, 0.166]	[−0.880, −0.214]	[−0.483, 0.244]
Ticket.ClassStandard			0.417**	0.516***	0.527***
			[0.134, 0.709]	[0.228, 0.813]	[0.227, 0.836]
Ticket.TypeAnytime			−0.623***	−0.702***	−0.714***
			[−0.867, −0.385]	[−0.951, −0.459]	[−0.980, −0.454]
Ticket.TypeOff-Peak			0.314**	0.330***	0.272**
			[0.125, 0.502]	[0.140, 0.520]	[0.071, 0.473]
Journey.StatusDelayed				−0.852***	−0.804***
				[−1.049, −0.658]	[−1.011, −0.598]
Reason.for.DelayStaff					0.180
					[−0.182, 0.534]
Reason.for.DelayStaffing					1.245***
					[0.937, 1.555]
Reason.for.DelayTechnical Issue					1.433***
					[1.175, 1.694]
Reason.for.DelayTraffic					0.763***
					[0.425, 1.099]
Reason.for.DelayWeather					−0.627***
					[−0.914, −0.343]
Num.Obs.	3333	3333	3333	3333	3333
AIC	3862.8	3549.2	3495.7	3421.7	3118.8
BIC	3893.3	3598.1	3562.9	3495.1	3222.7
Log.Lik.	−1926.387	−1766.585	−1736.843	−1698.873	−1542.391
F	3.877	44.912	35.072	35.364	36.421
RMSE	0.44	0.42	0.41	0.41	0.39

+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001

5.2. Assesment of Performance

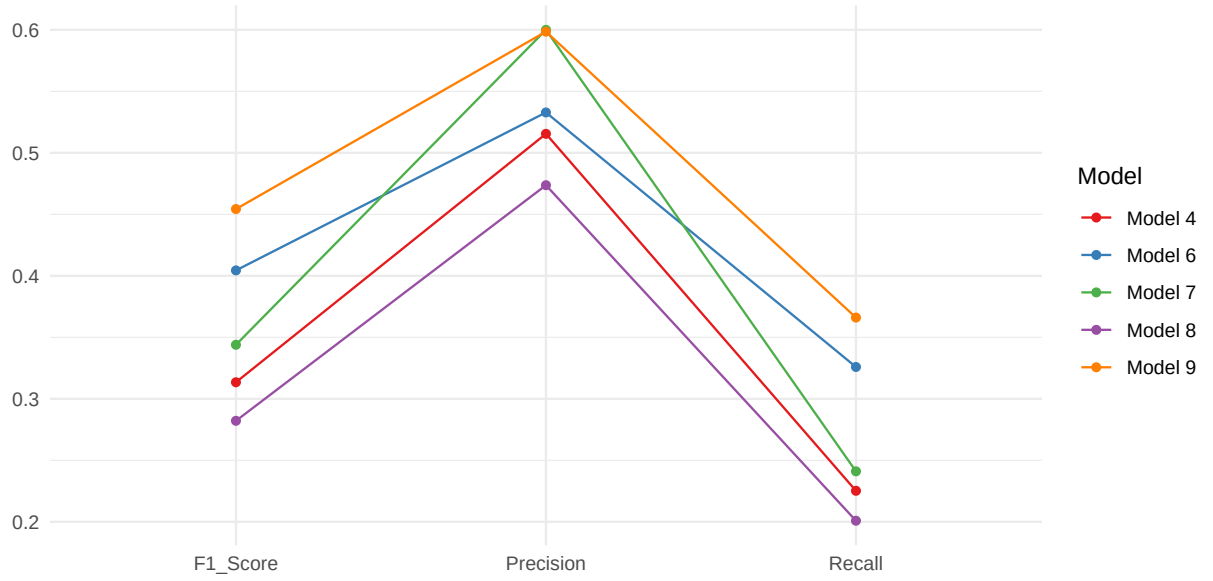
We analyse three metrics to assess performance:

- *Recall*: The proportion of actual positives correctly predicted.
- *Precision*: The proportion of predicted positives that are true positives.
- *F1 Score*: A balanced metric that combines precision and recall.

Our focus will be on minimizing False Negatives (FN), prioritizing *Recall* and *F1 Score*, which place greater emphasis on reducing FNs, since they may cause delays in responding to refund requests, as the model would underestimate the actual number.

Models 1, 2, 3, and 5 all predicted that no passengers in the testing set would request a refund, demonstrating a lower level of accuracy. Analysing the performance metrics, Model 9 will be selected, since appears to be the best-performing model, with the highest *Recall* and *F1 Score* values.

Graph 6: Performance Metrics



5.3. Interpretation of Results

For the passenger with the highest probability, Model 9 predicted an 88% chance of a refund request. This passenger paid £56, used an Adult railcard, had a Standard ticket class, purchased the ticket in advance, and experienced a cancelled journey due to a "technical issue".

$$\log(\text{Odds}) = -0.992 + 1.013 + 0.527 + 1.433 = 1.981$$

Thus, the value of the odds is equal to 7.25:

$$\frac{\text{P(Refund.Request)}}{1 - \text{P(Refund.Request)}} = e^{1.981} = 7.25$$

Finally, the predicted probability of requesting a refund was 88%:

$$P(\text{Refund.Request}) = \frac{7.25}{1 + 7.25} = 0.879$$

```
## [1] 0.8788308
```

The lowest probability is equal to 2%. This passenger paid £19, had no railcard, chose a "First Class" ticket class, purchased an "Anytime" ticket, and experienced a delayed journey because of bad weather.

$$\log(\text{Odds}) = -0.992 + 0.499 - 1.533 - 0.714 - 0.804 - 0.627 = -4.171$$

The value of the odds is equal to 0.015:

$$\frac{\text{P(Refund.Request)}}{1 - \text{P(Refund.Request)}} = e^{-4.171} = 0.015$$

The predicted probability of requesting a refund was 2%:

$$P(\text{Refund.Request}) = \frac{0.015}{1 + 0.015} = 0.015$$

```
## [1] 0.01521153
```

5.4. Predicting another data set

Finally, we determine the likelihood of passengers in *ToPredict* requesting a refund. We assign the value of zero for passengers with on-time journeys, and then predict the likelihood of requesting a refund for the rest. The results indicate that, with a threshold of 0.5, **none of the 8 passengers are likely to request a refund based on Model 9**:

```
##   predicted_probabilities refund_prediction
## 1               0.4325543                 0
## 2               0.2696997                 0
## 3               0.4274649                 0
## 4               0.3203494                 0
## 5               0.4325543                 0
## 6               0.1826526                 0
## 7               0.0000000                 0
## 8               0.0000000                 0
```

Since we focus on minimizing FNs, the threshold can be lowered to 0.4. With this adjustment, Model 9 would predict that 3 out of 8 passengers in *ToPredict* would request a refund.

```
##   predicted_probabilities NEWrefund_prediction
## 1               0.4325543                    1
## 2               0.2696997                    0
## 3               0.4274649                    1
## 4               0.3203494                    0
## 5               0.4325543                    1
## 6               0.1826526                    0
## 7               0.0000000                    0
## 8               0.0000000                    0
```