NEXT: Driver Classification Report

Tyler Adair

Executive Summary

Given the data detailing biographical, single-day, and cumulative information regarding truck owners, descriptive analysis and predictive modeling was performed to consistently classify high-performing drivers. Simple data analysis was done to identify the distributions of each feature, as well as to find any feature correlations. Data preprocessing was completed to remove irrelevant samples and features from the dataset through a data pipeline. Once the data was prepared, a plethora of models were trained on the training data, such as Logistic Regression, Ensemble Methods, TruncatedSVD, and Neural Networks. The models served not only to produce an effective finalized model, but to perform additional analysis of each model through hypothesis testing and dimensionality reduction. Lastly, K-Fold Cross Validation was performed on the Ensemble and Neural Network models to tune their hyperparameters. Upon prediction, it was found that the Random Forest Classifier model performed the best, producing an F1 score of 0.984 on the test data.

From the introductory data analysis, it was found that some features showed significant correlation to the manufactured labels on the training data, including the amount of marketplace loads and brokerage loads performed by the operator. Within these two features, the correlation matrix also showed that over the road (otr) brokage loads largely dominates the brokerage load

amount, indicating that the presence of brokers on ATLAS is small, while ATLAS largely correlates to the overall marketplace loads, indicating that drivers from this dataset that aren't assigned through brokers utilize ATLAS at a high rate.

Since the best performing model is an ensemble method, which makes ascertaining significant features of the model difficult to find, further analysis of the Logistic Regression model could be performed to identify key features that could lead to high performance in truck operators.

Background/Introduction

Any individual who travels by car can identify the importance and scale of the trucking industry. NEXT reports that the trucking industry is valued at $800 billion, with over half of that valuation coming from Full Truckloads (FTLs). In this age of trucking, many truckers are assigned jobs through brokers, which, along with losing money to broker commission, leads to less efficiency due to this indirect communication between shippers and carriers. Additionally, with the additional requirement of electric log devices (ELD), the industry will be further fragmented as the standards to be a trucker outweighs the low pay and benefits of the position.

The process of getting a shipment from port to warehouse to distribution center is linked by the truckers transporting between two of more of these possible locations. Within any of these steps, poor planning and communication can lead to inefficient jobs for truckers. This is where NEXT is stepping in with logistics solutions to the trucking industry.

The central challenge within the logistics applies to truckers and shippers alike: they want the most efficient and rewarding way to transport containers from location to location. This is made more complicated by the real-time needs for movement of containers to each possible location, including the considerations of load size, timing, job assignment, communication to fleet owners and truck operators, etc. Overall, this requires considering many variables in building an effective, robust solution that leads to positive consequences for all involved. By using the data collected from past shipments, a variety of applications in data science can be utilized to create a streamlined application that eases the difficulty of posting, assigning, executing, and reviewing jobs.

With that being said, there are many factors to analyze and improve, with one of those being the assignment of jobs. By collecting data on past jobs, one can identify truck or fleet operators that output higher performance than others. This can lead to a variety of ideas and expansions, such as priority assignment of drivers to longer, more difficult jobs, linking different assignments together based on location of loads, etc. The purpose of this experiment was to analyze and build models that could consistently identify high-performing drivers, which would be a useful system in developing a larger-scale idea like the ones mentioned above.

Methodology

Given the dataset, the features were first organized and understood via simple descriptive metrics, which are shown below:

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 84414 entries, 0 to 999
Data columns (total 31 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   Unnamed: 0                 84414 non-null  int64
 1   dt                         84414 non-null  object
 2   weekday                    84414 non-null  object
 3   year                       84414 non-null  int64
 4   id_driver                  84414 non-null  int64
 5   id_carrier_number          84414 non-null  object
 6   dim_carrier_type           84414 non-null  object
 7   dim_carrier_company_name   84365 non-null  object
 8   home_base_city             84369 non-null  object
 9   home_base_state            84369 non-null  object
 10  carrier_trucks             84414 non-null  object
 11  num_trucks                 84344 non-null  float64
 12  interested_in_drayage      84414 non-null  object
 13  port_qualified             84414 non-null  object
 14  signup_source              84414 non-null  object
 15  ts_signup                  84414 non-null  object
 16  ts_first_approved          71978 non-null  object
 17  days_signup_to_approval    71978 non-null  float64
 18  driver_with_twic           84414 non-null  object
 19  dim_preferred_lanes        3451 non-null   object
 20  first_load_date            84414 non-null  object
 21  most_recent_load_date      83414 non-null  object
 22  load_day                   84414 non-null  object
 23  loads                      84414 non-null  int64
 24  marketplace_loads_otr      84414 non-null  int64
 25  marketplace_loads_atlas    84414 non-null  int64
 26  marketplace_loads          84414 non-null  int64
 27  brokerage_loads_otr        84414 non-null  int64
 28  brokerage_loads_atlas      84414 non-null  int64
 29  brokerage_loads            84414 non-null  int64
 30  total_loads                83414 non-null  float64
dtypes: float64(3), int64(10), object(18)
memory usage: 20.6+ MB
None
```

```
         num_trucks  days_signup_to_approval         loads  \
count  84344.000000             71978.000000  84414.000000
mean      22.597185               298.803190      2.075473
std       48.840386               390.414603      2.666080
min        1.000000                 0.000000      1.000000
25%        1.000000                 0.000000      1.000000
50%        4.000000                61.000000      1.000000
75%       14.000000               497.000000      2.000000
max      195.000000              1653.000000    129.000000

       marketplace_loads_otr  marketplace_loads_atlas  marketplace_loads  \
count           84414.000000             84414.000000       84414.000000
mean               29.491447                71.547326         101.038773
std                88.274149               194.479172         214.501677
min                 0.000000                 0.000000           0.000000
25%                 0.000000                 0.000000           0.000000
50%                 2.000000                 0.000000          13.000000
75%                23.000000                18.000000          94.000000
max               902.000000              1324.000000        1348.000000

       brokerage_loads_otr  brokerage_loads_atlas  brokerage_loads  \
count         84414.000000           84414.000000     84414.000000
mean            148.160222              13.077381       161.237603
std             415.462234              42.267832       413.278914
min               0.000000               0.000000         0.000000
25%               0.000000               0.000000         5.000000
50%              15.000000               0.000000        37.000000
75%             110.000000               1.000000       135.000000
max            4266.000000             371.000000      4266.000000

              label
count  84414.000000
mean       0.123119
std        0.328576
min        0.000000
25%        0.000000
50%        0.000000
75%        0.000000
max        1.000000
```

Figure 1: Info and descriptive statistics on dataset features.

Figure 2: Distribution of continuous features.

Additionally, labels were manufactured for the training data, with high-performing drivers assigned a label of 1 if they were in 75th percentile of total loads and most recent load date.

Next, to identify any possible feature correlations, a correlation matrix was produced, which produced the following results:

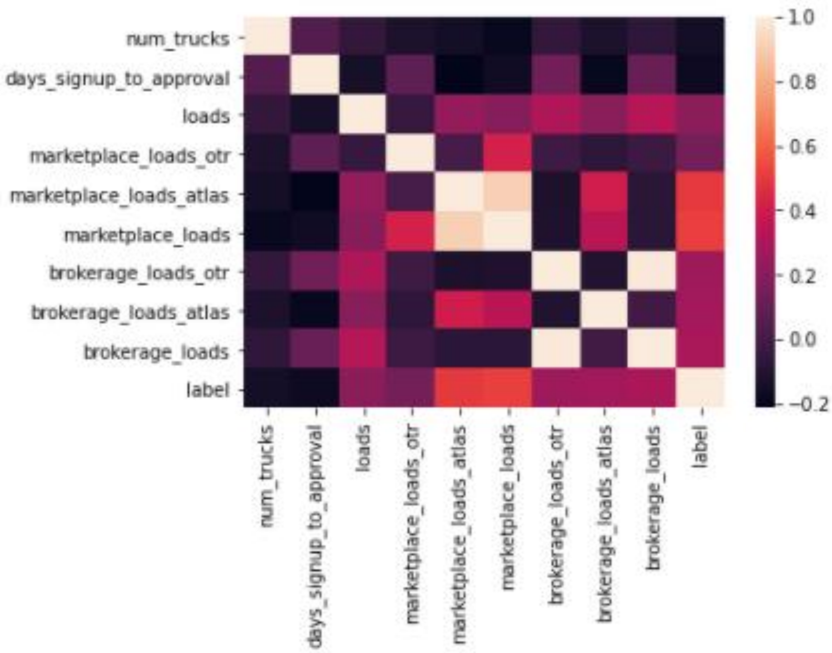

Figure 4: Correlation matrix of the features. The lighter the color, the more positive the correlation between two features.

	num_trucks	days_signup_to_approval	loads
num_trucks	1.000000	0.051573	-0.048151
days_signup_to_approval	0.051573	1.000000	-0.136858
loads	-0.048151	-0.136858	1.000000
marketplace_loads_otr	-0.120255	0.085015	-0.034094
marketplace_loads_atlas	-0.150301	-0.213596	0.237985
marketplace_loads	-0.185767	-0.159449	0.201740
brokerage_loads_otr	-0.049881	0.133193	0.312709
brokerage_loads_atlas	-0.120857	-0.188274	0.207638
brokerage_loads	-0.062506	0.109355	0.335597
label	-0.147923	-0.173430	0.216259

	marketplace_loads_otr	marketplace_loads_atlas
num_trucks	-0.120255	-0.150301
days_signup_to_approval	0.085015	-0.213596
loads	-0.034094	0.237985
marketplace_loads_otr	1.000000	0.011548
marketplace_loads_atlas	0.011548	1.000000
marketplace_loads	0.422001	0.911408
brokerage_loads_otr	-0.013237	-0.119366
brokerage_loads_atlas	-0.064892	0.408966
brokerage_loads	-0.019944	-0.078170
label	0.150293	0.499873

	marketplace_loads	brokerage_loads_otr
num_trucks	-0.185767	-0.049881
days_signup_to_approval	-0.159449	0.133193
loads	0.201740	0.312709
marketplace_loads_otr	0.422001	-0.013237
marketplace_loads_atlas	0.911408	-0.119366
marketplace_loads	1.000000	-0.113671
brokerage_loads_otr	-0.113671	1.000000
brokerage_loads_atlas	0.344087	-0.102387
brokerage_loads	-0.079081	0.994811
label	0.515063	0.268878

	brokerage_loads_atlas	brokerage_loads	label
num_trucks	-0.120857	-0.062506	-0.147923
days_signup_to_approval	-0.188274	0.109355	-0.173430
loads	0.207638	0.335597	0.216259
marketplace_loads_otr	-0.064892	-0.019944	0.150293
marketplace_loads_atlas	0.408966	-0.078170	0.499873
marketplace_loads	0.344087	-0.079081	0.515063
brokerage_loads_otr	-0.102387	0.994811	0.268878
brokerage_loads_atlas	1.000000	-0.000654	0.279523
brokerage_loads	-0.000654	1.000000	0.298887
label	0.279523	0.298887	1.000000

Figure 5: Raw numerical form of Figure 4.

This correlation matrix showed a couple of interesting distinctions. First, the manufactured label seems to have a decently significant correlation with marketplace loads, both cumulatively and ones covered by drivers using ATLAS. Secondly, marketplace loads and brokerage loads are heavily correlated to ATLAS-based and OTR-based loads, respectively. While this feature correlation is evident, the features used to find cumulative marketplace and brokerage loads were kept to explore the significance of these individual parts.

Following this, data preprocessing was performed. Most of the features removed described single-day, biographical, and irrelevant details related to the task of classifying high-performing drivers. Then, null values were dealt with in two ways. Columns having very few null values had the rows with those null values removed, while more significant occurrences of null were median imputed. Categorical variables were one-hot encoded, numerical variables were standardized, and a cross term was produced between the number of trucks and whether an operator was a truck driver or a fleet owner.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 84300 entries, 0 to 999
Data columns (total 19 columns):
 #   Column                         Non-Null Count  Dtype
---  ------                         --------------  -----
 0   home_base_state                84300 non-null  object
 1   carrier_trucks                 84300 non-null  object
 2   num_trucks                     84300 non-null  float64
 3   interested_in_drayage          84300 non-null  object
 4   port_qualified                 84300 non-null  object
 5   days_signup_to_approval        84300 non-null  float64
 6   driver_with_twic               84300 non-null  object
 7   loads                          84300 non-null  int64
 8   marketplace_loads_otr          84300 non-null  int64
 9   marketplace_loads_atlas        84300 non-null  int64
 10  marketplace_loads              84300 non-null  int64
 11  brokerage_loads_otr            84300 non-null  int64
 12  brokerage_loads_atlas          84300 non-null  int64
 13  brokerage_loads                84300 non-null  int64
 14  label                          84300 non-null  int64
 15  dim_carrier_type_Fleet         84300 non-null  float64
 16  dim_carrier_type_Owner Operator  84300 non-null  float64
 17  num_trucks_fleet               84300 non-null  float64
 18  num_trucks_operator            84300 non-null  float64
dtypes: float64(6), int64(8), object(5)
memory usage: 12.9+ MB
None
```

Figure 6: Information on remaining features prior to one-hot encoding, standardization, and production of cross term.

After the data was processed and prepared to be inputted into models, a variety of models were fitted and tested on the data. A Logistic Regression model was made as a base classifier, which was then bootstrapped to get a t-value and p-value for the model. Since the preprocessing resulted in the dataset having 71 features, TruncatedSVD was performed as a dimensionality reduction technique on the features, resulting in 5 features that best capture the variance of the data. Following this, the actual predictive models were produced. A random forest classifier and a two-layer neural network was trained, with its hyperparameters being tuned via K-Fold Cross

Validation. Once they were optimized, the models were used to predict the scores on the test data.

Results

The results of the introductory data analysis are detailed in Figures 1-6, where feature correlations and distributions are shown and discussed.

When running the random forest classifier, neural network, and results of k-fold cross validation on both of those models, the following results were produced:

	R2 Score	**F1 Score**
Random Forest Classifier (no optimization)	1.0	1.0
Neural Network (no optimization)	0.984	0.9334
Random Forest Classifier (K-Fold Cross Validation)	1.0	1.0
Neural Network (K-Fold Cross Validation)	0.987	0.9484
Neural Network (K-Fold CV, Optimization for activation and optimizer)	0.9927	0.9712

Figure 7: Scores from training different models. (scores come from predicting on validation set)

	F1 Score
Neural Network	0.732
Random Forest Classifier	0.984

Figure 8: Test scores on scores.csv. (Using best NN and Random Forest models)

Discussion

For training the Random Forest and Neural Network models, the scores were improved by tuning the hyperparameters of each model. For the random forest, the max depth of each tree in the ensemble, along with the number of estimators, were tuned, with the best model having a max depth of 16 with 100 estimators. For the neural network, the learning rate, activation function, and optimizer was tuned, with the best model having a learning rate of 1e-3, tanh activation, and adam for its optimizer. Further optimization could have been performed on the neural network by modifying its architecture, tuning the parameters within the adam optimizer, and trying a decaying learning rate with more iterations to train.

The one downside to these models is their interpretability. Since a random forest uses bagging in its training, it is not easy to identify which features were most critical in producing good results. Further analysis could be done to identify variable importance, but other methods, like using Logistic Regression, are useful in finding these variables. The neural network uses many neurons to train, which makes finding direct variable importance tough, since each variable's impact is spread across many neurons.

Based on these findings, there are many avenues to traverse that capitalize on these results. From the introductory data analysis, since the manufactured labels showed correlation to all load-based metrics, doing deeper analysis on these features might shed some light on their impact and relation to high-performing drivers. More information regarding these loads would be needed, but could prove fruitful in other applications of planning and assigning jobs. Additionally, if successful models are used to predict high-performing drivers, doing more analysis on these high-performing drivers, including determining more factors in their success, could be a catalyst to help all drivers' performances improve. Lastly, doing separate analyses on fleet owners and truck owners might show some different results in terms of their performances. Hypothetically, fleet owners might be held to a higher standard of production than individual truck operators, but would need more analysis to prove true.

Another cause of concern is the distribution of the given dataset in relation to the population of shipments done in the western U.S. If the sample data is not representative of the larger distribution, these models might not generalize well if the trucker base scales larger and larger.

Conclusion

Overall, statistical analysis was performed on the dataset, followed by predictive model-building through data preprocessing, model development, analysis, and optimization, and testing. This experiment revealed key distinctions in predicting high-performing drivers, such as marketplace and brokerage load counts and distributions. Models were built and optimized to produce good results on the test data, with room for improvement. Deeper analysis could be performed on these load counts, as well as on fleet owners and individual truck owners separately, to provide a deeper insight to improve the experience of truckers.