Predicting service operations from connected car telematics

Nicholas Hirons
Julian Kudszus
Soham Kudtarkar
Spencer Lee

CARFORCE

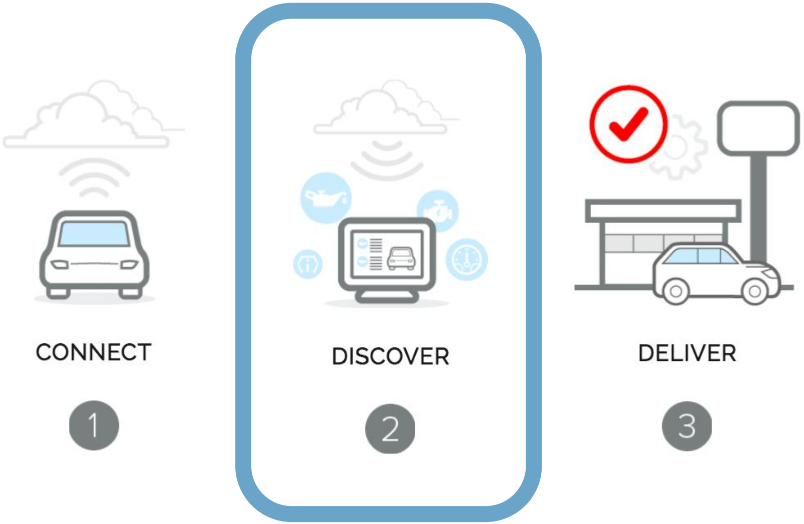
Mentored by:
Jessika Lora
CarForce
thecarforce.com

The problem

- There are over 165 million smart capable but disconnected cars
- CarForce leverages connected car telematics / data
- Provides dealerships with real time updates on the health of their customers' vehicles

The problem

- **Our goal** is closely related: predict required service operations from connected car data
- **Service providers** can know ahead of time the most likely operations to manage their resources efficiently
- **Consumers** could receive a transparent prediction of how their car may be serviced
- Improve trust and customer satisfaction



Timeline

Mid Sep

Form group

Initial members: Julian and Spencer. Nicholas and Soham joined after initial placement.

Early Oct

Engage with mentor

Discuss getting data and directions for project

Late Oct

Discuss project scope before receiving data

Favored idea: predicting future engine issues from engine error codes

Early Nov

Discuss alternative projects

Due to delays in receiving data, examine other possible related projects and datasets

Mid Nov

Receive data

Low tech project demo
Conduct initial EDA
Decide on project scope
Clean and process data

Early Dec

Final results and presentation

Finish model selection/training
Analyze results
Complete presentation/report

Background on data

- Raw, anonymized service record database
- ~560k rows and ~180 columns
- Many missing values & irrelevant columns
- ~16k with engine error codes
    - We used this subset of data for training

Feature selection: outputs

- Logical choice: operation code descriptions
- Potential to help both service providers AND consumers:
  - Service providers learn ahead of time what needs to be examined
  - Consumers gain insight into how car may be serviced
- **Challenge:** Categorical/numeric to text mapping is uncommon.

Feature selection: outputs

Example of raw operation code descriptions

REPLACE 8 INJ, HIGH ANDLOW PRESS FUEL PUMPS, FULLINES AND RAILS, RELINEFUEL TANK, FLUSH FUELSYSTEM, LOF, NEW COOLANT|||PERFORM MULTI-POINTINSPECTION|BATTERY TEST PERFORMEDAND BATTERY OK ON THISVISIT|BATTERY TERMINALS GOOD ATTHIS TIME|TIRES INSPECTED AND OK ONTHIS VISIT.||SEE LINE A|DRAW TIME|DRAW TIME

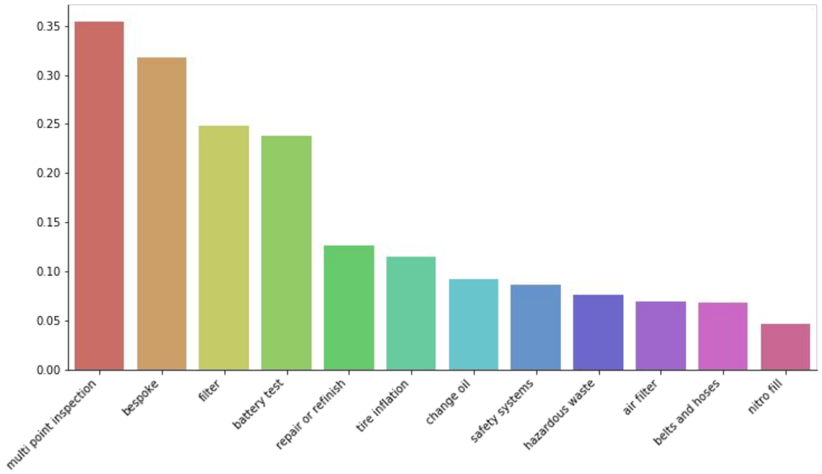
- Filled with merged words and spelling errors
- Focused on separating merged words using dynamic programming

Feature selection: outputs

- From cleaned and separated words, manually selected most common operations from common words/pairs

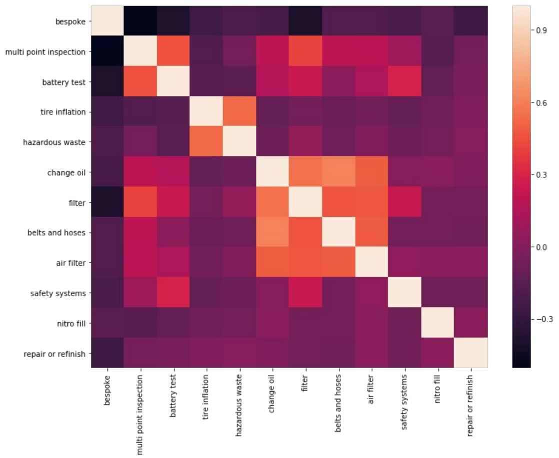
- Operation codes without any common operations were mapped to ‘bespoke’

- Multi-label classification problem (rather than one vs. all)

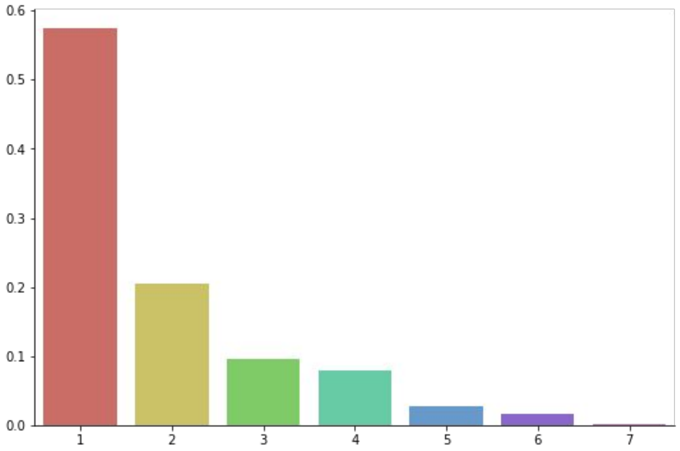


Feature selection: outputs

Correlation heatmap of output variables



Frequency of output variables per string



Timeline
Background on data
Feature selection
Model selection
Results

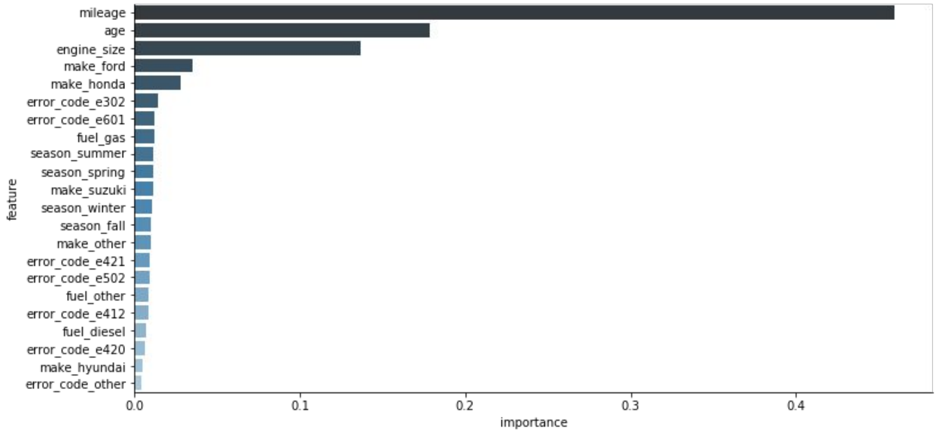
Feature selection: inputs

- Priority was to use engine error codes, leveraging the ‘connected car’
- Other features were selected based on simplicity, availability and our intuition of predictive power
- Categoricals were one-hot-encoded, capped at 5-7 categories with the rest mapped to ‘other’
    - Based on early random forest prototyping, using more dummy variables did not improve performance

Error Code	Make	Engine Size	Fuel Type	Mileage	Car Age	Season

Feature selection: inputs

Feature importance on baseline random forest for multi-point inspection



Model selection

- Comparison of classification algorithms: Random Forests, kNN, SVM, Logistic Regression
- Given output class imbalance, F-score was used as the main metric for CV
- Models were trained across a parameter grid on each output variable using 5-fold cross validation
- Results were very mixed, but k-NN consistently outperformed and provided an interesting result!

Results

k-Nearest Neighbors
The only output feature with strong results is arguably the most important - **whether your car requires any repair or refinish!**

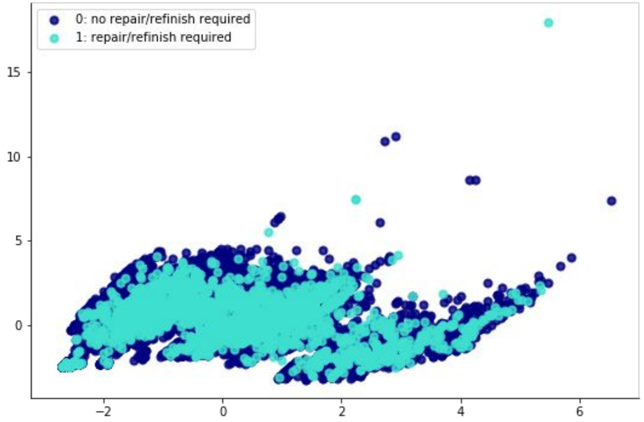
Only 13% ground truth
0.94 F-score

output	accuracy	precision	recall	f-score
bespoke	0.57	0.02	0.04	0.02
multi point inspection	0.61	0.12	0.31	0.17
battery test	0.68	0.10	0.19	0.13
tire inflation	0.78	0.10	0.09	0.10
hazardous waste	0.82	0.12	0.07	0.09
change oil	0.80	0.10	0.08	0.09
filter	0.67	0.09	0.18	0.12
belts and hoses	0.81	0.05	0.02	0.03
air filter	0.82	0.14	0.08	0.10
safety systems	0.80	0.05	0.03	0.04
nitro fill	0.84	0.17	0.06	0.09
repair or refinish	0.98	0.93	0.94	0.94

Results: Why k-NN may have worked

- Classes don't appear to be very separable
- However, it makes sense that there are neighborhoods / clusters of similar cars
- An algorithm that is sensitive to the local structure of data, like k-NN, may therefore be the most effective for this application

First two components of PCA



Results

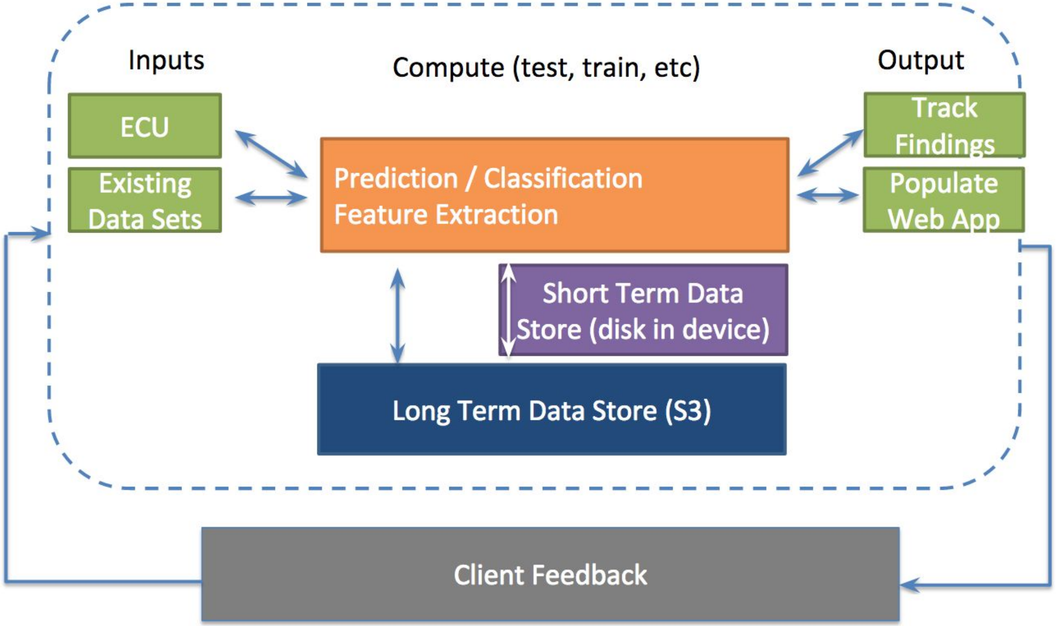
- In the current problem formulation, engine error codes were not as predictive as hypothesized
- Service records may not be in a consistent format between different dealerships / owners
    - Challenging to build models that generalize
- Dealerships may not yet have an informed approach to engine error codes
    - Historical data of limited use until dealerships gain better understanding of

Extension: Training separate models for each make

- Currently examining effectiveness of training similar models on each make
- Rather than using make as a one-hot-encoded feature, use the subset of the data for training
- This will allow us to capture more information from the 'tail-end' of the make distribution

Demonstration of working code

Architecture



Intended user interface

- Simple, easy to understand web interface
    - Vehicle engineering can be complex, so an easy-to-understand interface that avoids ambiguities is ideal
- Different dashboard between service provider and consumer (if available to consumer)
- The interface should take into account the important features that the model requires to make a good guess on potential issues and need for repair

Car Service Predictions

Error Code

E101

Make

Chevrolet

Engine Size (L):

1 3.2 10

1 1.9 2.8 3.7 4.6 5.5 6.4 7.3 8.2 9.1 10

Fuel Type

Diesel

Mileage

0 50,000 300,000

0 30,000 60,000 90,000 120,000 150,000 180,000 210,000 240,000 270,000 300,000

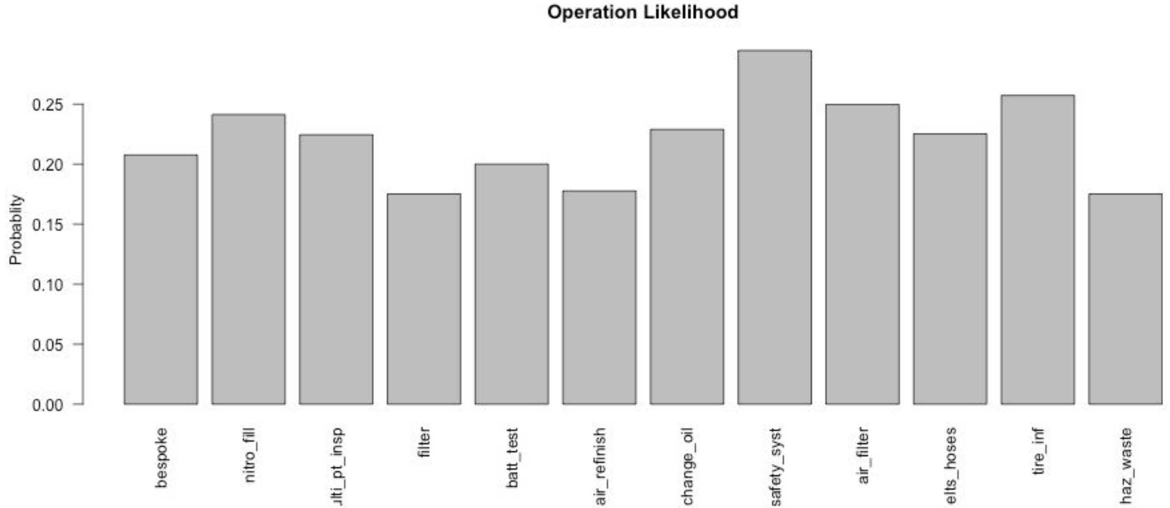
Car Age

0 5 25

0 3 6 9 12 15 18 21 24 25

Date:

2017-12-04



It is highly unlikely (18%) that your car will need some repair or refinishing. Your dealership will reach out to you soon. If you do not hear from them within 7 days, please contact them.

Car Service Predictions

Error Code

E420

Make

Honda

Engine Size (L):

1 1.9 10

1 1.9 2.8 3.7 4.6 5.5 6.4 7.3 8.2 9.1 10

Fuel Type

Gas

Mileage

0 237,887 300,000

0 30,000 60,000 90,000 120,000 150,000 180,000 210,000 240,000 270,000 300,000

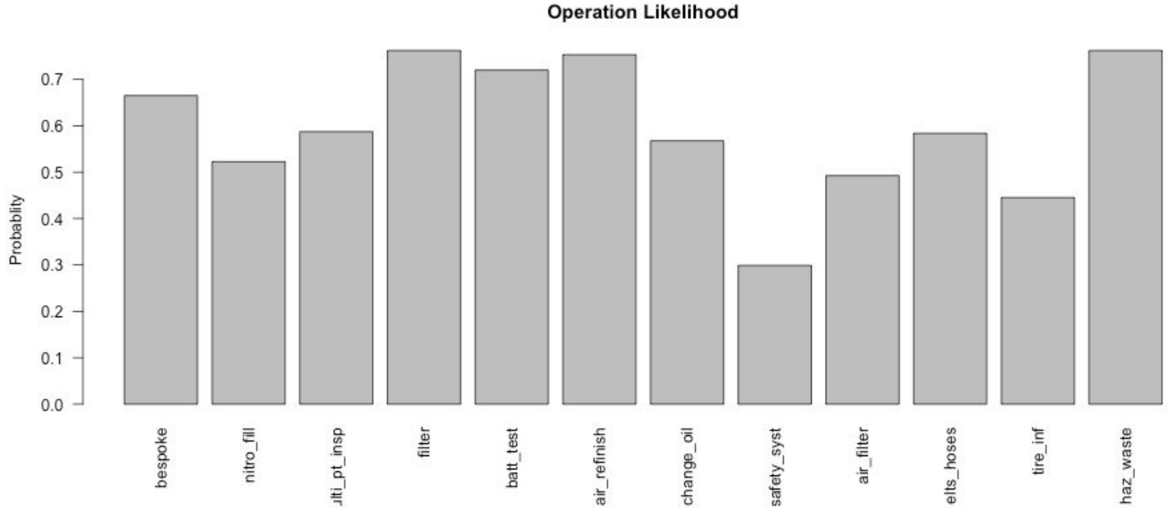
Car Age

0 18 25

0 3 6 9 12 15 18 21 24 25

Date:

2017-10-10



It is highly likely (75%) that your car will need some repair or refinishing. Your dealership will reach out to you soon. If you do not hear from them within 7 days, please contact them.

Our learning path

- In the beginning, much of it was theoretical since we did not have data to work with
  - We brainstormed different approaches in the case that we didn't get the data, including cooking our own data based on a similar distribution

- When we did get data, it seemed as though we had a lot to work with: about 560k rows and 180 columns from two dealerships
  - Unfortunately, only ~16k of those rows had usable codes

- After wrangling and analyzing the data, we found that it was not as usable in the way that we initially believed, which led us to conclusions on the usability of the data that we had
  - Instead, we looked to more directly attainable goals with the data that we had

- **Overall, we were able to leverage the skills we gained in class to effectively run through the data analysis process and gain insight from what we found**

Further areas of interest

- Improving predictive power of current problem for services **beyond repair/refinish**
  - More information on engine error codes
  - Access to more systematic records of service and repair
- Predicting revenues and costs of service and repair from connected car data
  - Gain understanding of highest margin opportunities
  - Improve retention with most profitable customers
- Predicting future engine issues from past / current car data

Questions?

Link to GitHub repo

https://github.com/nhirons/carforce