Driving Style Signatures: Who's Behind the Steering Wheel?

(Student: Vinh Luong - 442069)

Introduction

The field of automotive insurance has a number of important questions regarding individuals' driving behaviors:

i. What data features are necessary to characterize a person's driving habits? - for one thing, such features can be used to appropriately price accident risk as well as cross-sell related insurance products;

ii. In the case of an accident claim, given such data features, how well can we verify whether the insured person - and not another person - is really behind the steering wheel when the incident occurs?

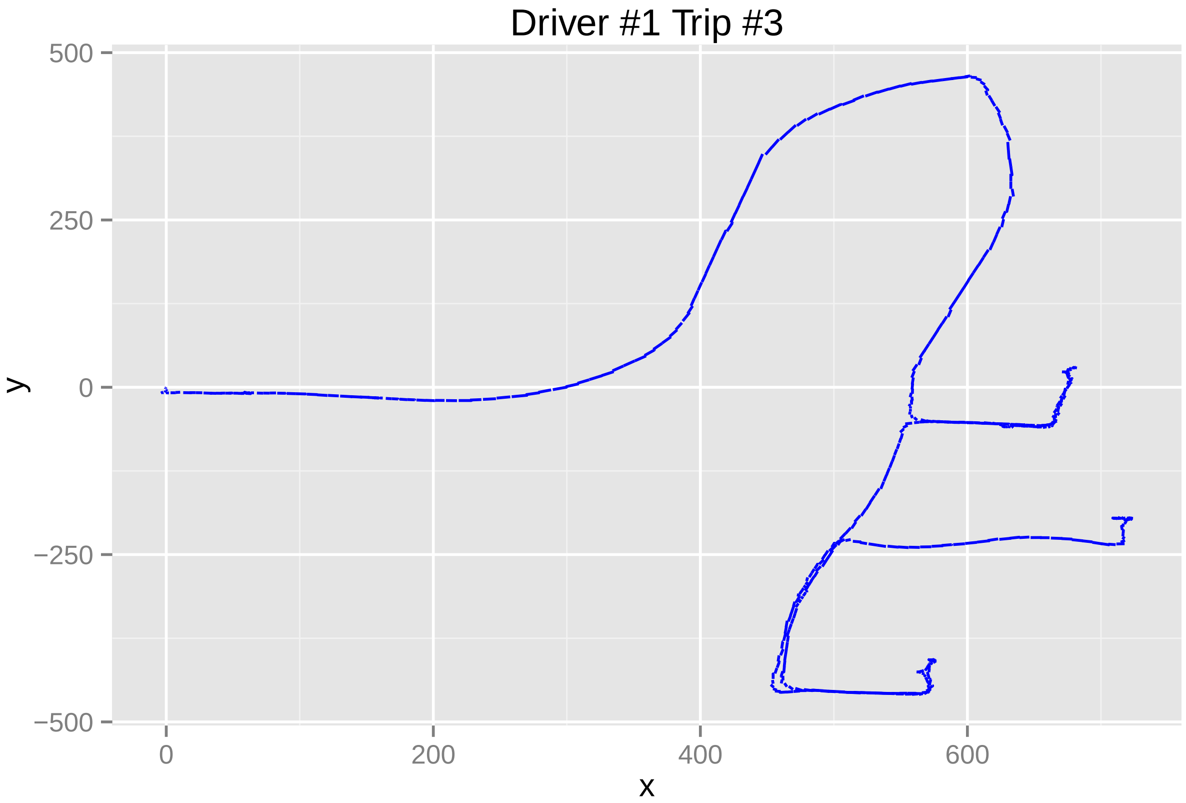
The advent of vehicle-mounted telematics devices has provided rich new data sources to address these issues. In this project, we attempt to develop a method to detect different people's own driving style "signatures" from a series of second-by-second GPS coordinate readings from their cars' telematics. We will show that by using just simple features such as velocity, acceleration, jerk, angular velocity and angular acceleration, we could already identify with about 70-80% accuracy whether the insured driver is driving his/her car.

1. Data and Data Preprocessing

1.1. Raw Data and Processed Higher-Order Features

We obtained second-by-second GPS (x_t, y_t) coordinate data from over half a million anonymized driving trips (200 trips by each of over 2,700 individual drivers) from French insurer AXA's Kaggle competition data set. This large dataset occupies nearly 6 GB of digital storage space once unpacked.

One trip is depicted below:



For anonymization purposes, each trip's starting point is centered at (0, 0) and the subsequent coordinates are rotated by a random angle.

From the raw (x_t, y_t) data, we derived a number of higher-order features, measured at every second t of each driving trip, as follows:

$\boldsymbol{x}$-velocity: $\Delta x_t = x_t - x_{t-1}$

$\boldsymbol{y}$-velocity: $\Delta y_t = y_t - y_{t-1}$

absolute velocity magnitude: $v_t = \left\| \begin{array}{c} \Delta x_t \\ \Delta y_t \end{array} \right\|$

$\boldsymbol{x}$-acceleration: $\Delta\Delta x_t = \Delta x_t - \Delta x_{t-1}$

$\boldsymbol{y}$-acceleration: $\Delta\Delta y_t = \Delta y_t - \Delta y_{t-1}$

signed acceleration magnitude: $a_t = \frac{1}{v_t} \left\langle \begin{bmatrix} \Delta\Delta x_t \\ \Delta\Delta y_t \end{bmatrix}, \begin{bmatrix} \Delta x_t \\ \Delta y_t \end{bmatrix} \right\rangle$

(i.e. acceleration in the direction of the velocity vector $\begin{bmatrix} \Delta x_t \\ \Delta y_t \end{bmatrix}$)

absolute acceleration magnitude: $|a_t|$

$\boldsymbol{x}$-jerk: $\Delta\Delta\Delta x_t = \Delta\Delta x_t - \Delta\Delta x_{t-1}$

$\boldsymbol{y}$-jerk: $\Delta\Delta\Delta y_t = \Delta\Delta y_t - \Delta\Delta y_{t-1}$

signed jerk magnitude: $j_t = \left\langle \begin{bmatrix} \Delta\Delta\Delta x_t \\ \Delta\Delta\Delta y_t \end{bmatrix}, \begin{bmatrix} \Delta\Delta x_t \\ \Delta\Delta y_t \end{bmatrix} \right\rangle \Big/ \left\| \begin{array}{c} \Delta\Delta x_t \\ \Delta\Delta y_t \end{array} \right\|$

(i.e. jerk in the direction of the acceleration vector $\begin{bmatrix} \Delta\Delta x_t \\ \Delta\Delta y_t \end{bmatrix}$)

absolute jerk magnitude: $|j_t|$

angle: $\theta = \arctan(\Delta y_t, \Delta x_t)$

signed angular velocity: $\Delta\theta_t = \theta_t - \theta_{t-1}$

absolute angular velocity: $|\Delta\theta_t|$

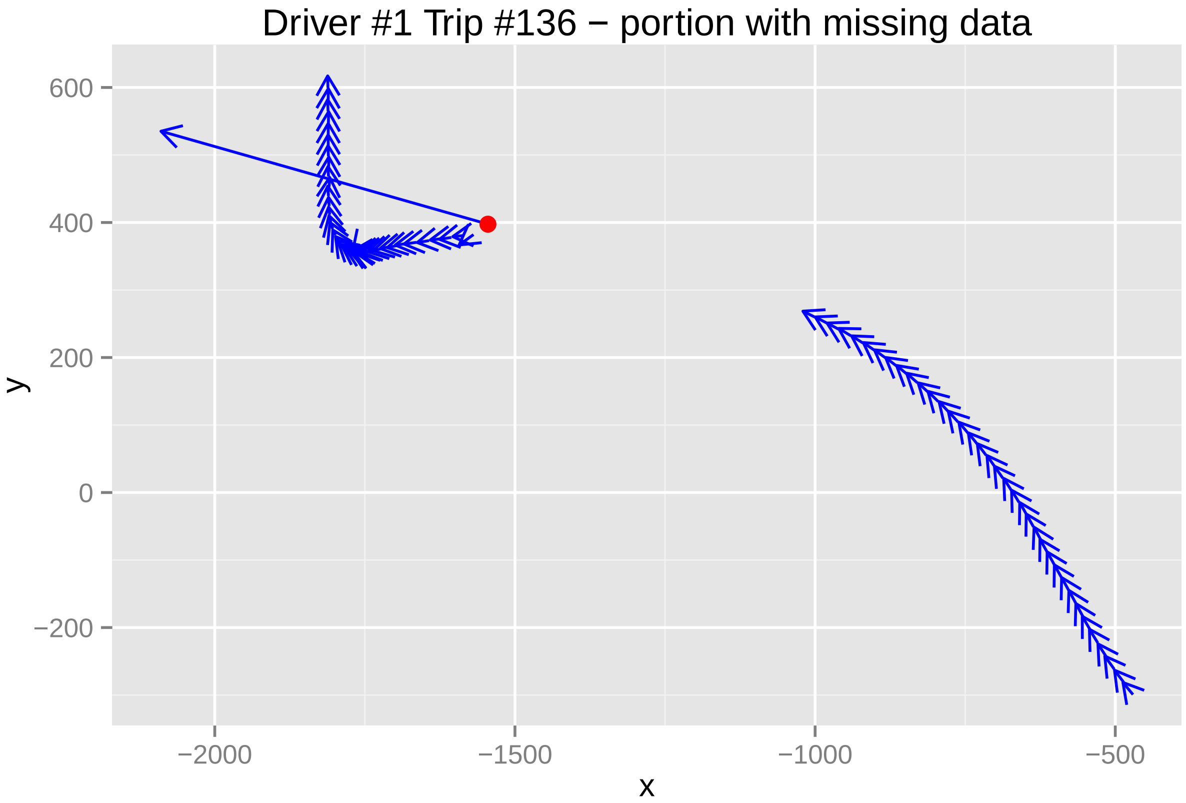
signed angular acceleration: $\Delta\Delta\theta_t = \Delta\theta_t - \Delta\theta_{t-1}$

absolute angular acceleration: $|\Delta\Delta\theta_t|$

Note that because the coordinates (x_t, y_t) and the angles θ_t are already randomly shifted and rotated by the data provider, they will not be of any value in the subsequent modeling task. Only the variables signifying the *rates of change* will be considered.

1.2. Data Cleaning

Before we could proceed with analyzing this large data set, we had to attend to some serious data integrity issues. It turns out that due to lost data transmission signals and/or some extreme anonymization measures, numerous raw driving trip data sets are plagued with coordinate "jumps", i.e. missing chunks of GPS readings. One example is portrayed below:

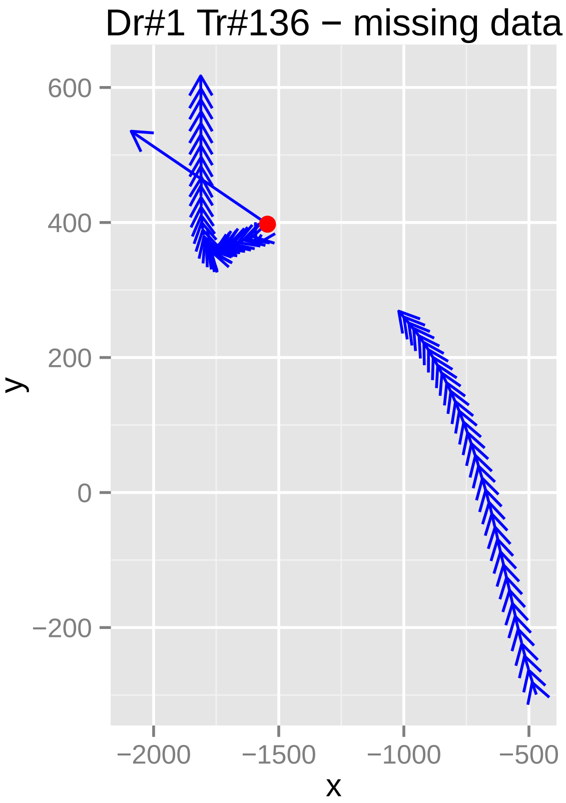


This problem is very pervasive, present in over 40,000 driving trips by almost all of the over 2,700 drivers in the database. Only eight drivers seem to have no missing data.

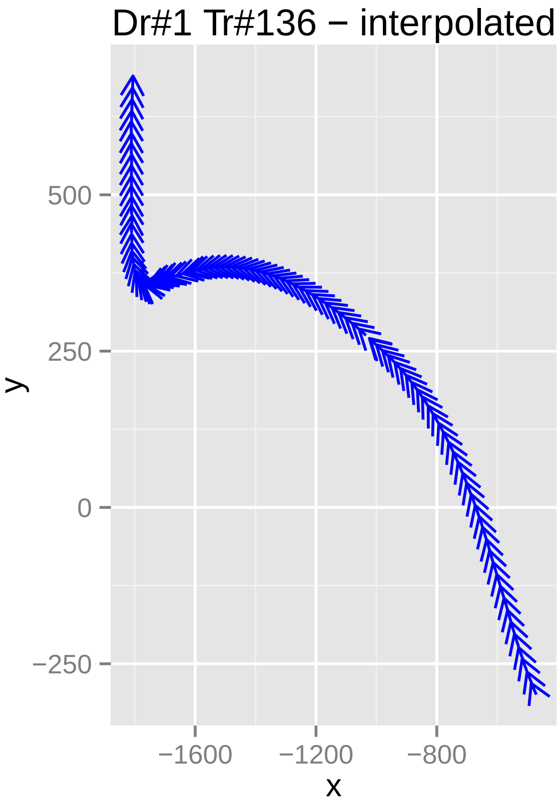
In the above depicted case, as well as in other missing data cases, the corrupt data portions - highlighted by red dots in the plots - manifest themselves quite apparently by an unreasonably large distance from (x_{t-1}, y_{t-1}) to (x_t, y_t), or equivalently an unreasonably high velocity v_t estimated from such consecutive pairs of coordinates. We hence devised a method to detect and interpolate the missing data, described at a high level as follows:

- detect time instances t with derived velocity $v_t > 36$ meters/second, which is approximately 80 miles/hour, the upper bound of the U.S. speed limits, or, equivalently, 130 kilometers/hour, the upper bound of the European speed limits - when one's car has a telematics device mounted, one should be quite properly incentivized not to over-speed!;
- look at time windows of 3 seconds before and 3 seconds after each of such instance t, and estimate the average velocities $v_{\text{before } t}$ and $v_{\text{after } t}$ and angular directions $\theta_{\text{before } t}$ and $\theta_{\text{after } t}$;
- by certain polygonal approximations, estimate the length of one or several smooth ***parabolic*** arcs spanning the locations $(x_{\text{before } t}, y_{\text{before } t})$ and $(x_{\text{after } t}, y_{\text{after } t})$ and with tangents at angles $\theta_{\text{before } t}$ and $\theta_{\text{after } t}$ at those points - it will become apparent in certain visualizations below why parabolic curves are more natural than straight lines or circular curves;
- estimate the number of seconds the vehicle needs to take to traverse such parabolic arc(s) at velocity $v_{\text{mean}} = \frac{1}{2}(v_{\text{before } t} + v_{\text{after } t})$; and
- interpolate missing intermediate locations along the parabolic arc(s), with certain technical adjustments to make the vehicle accelerate or decelerate evenly from $v_{\text{before } t}$ to $v_{\text{after } t}$.

With such a data interpolation method, the above case of Driver #1's Trip #136 could be corrected to the following:

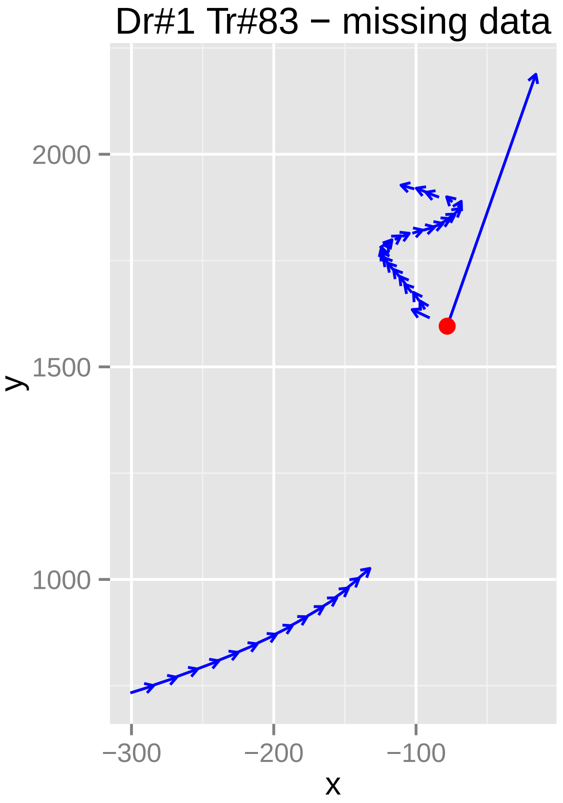


Dr#1 Tr#136 − missing data

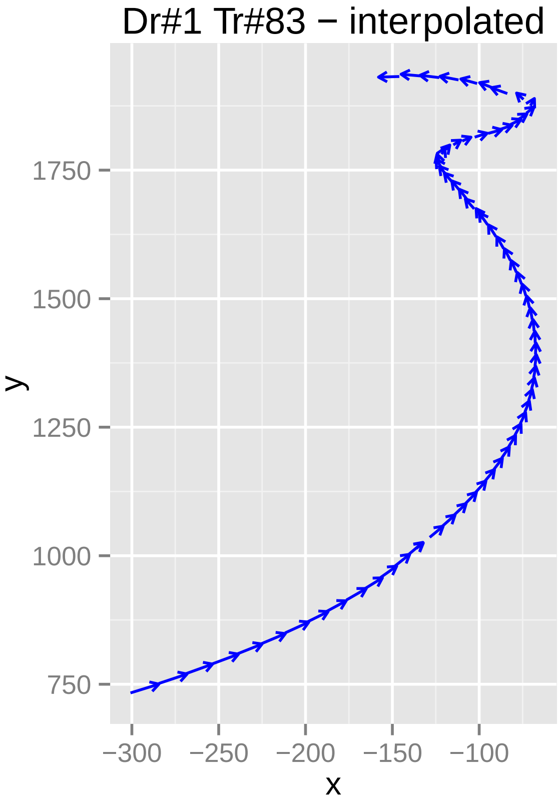


Dr#1 Tr#136 − interpolated

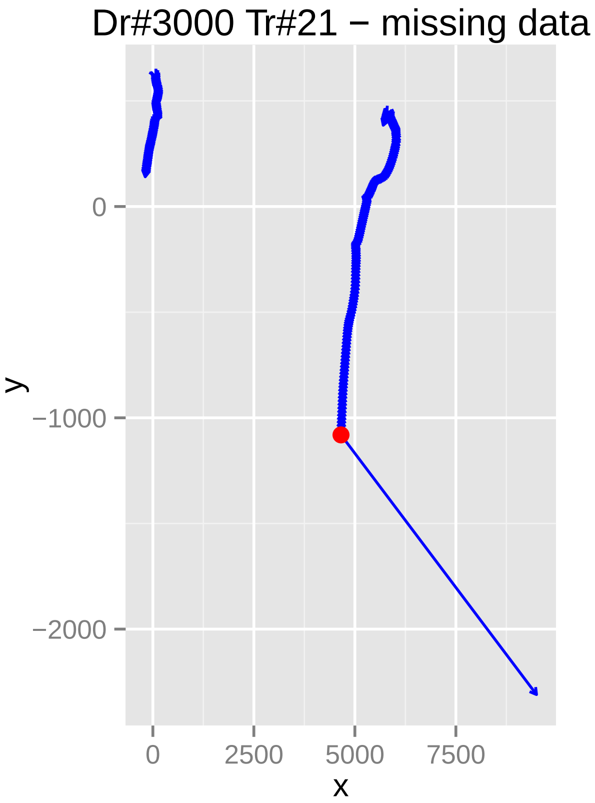
Below are several other examples demonstrating the efficacy of this method in recovering smooth, realistic-looking paths to replace missing data. Notice how parabolic-curve approximation works really well, while using straight lines or circular arcs would have created much less believable trajectories.

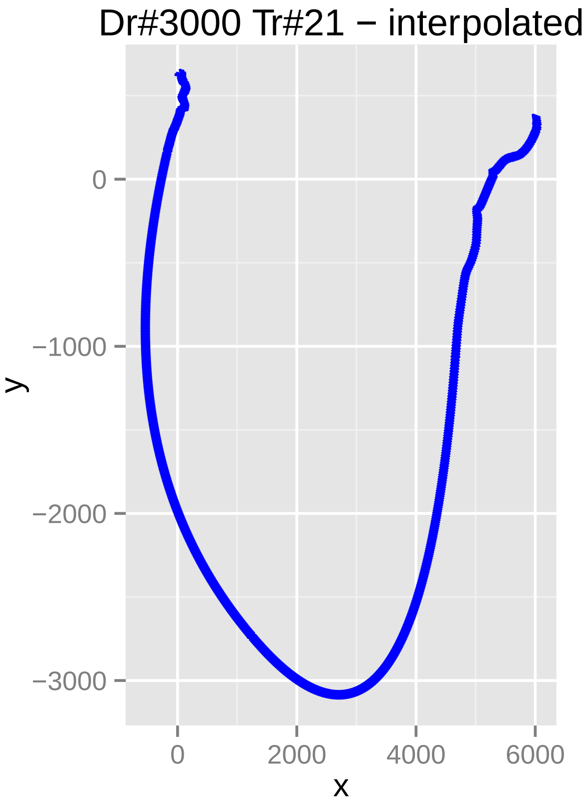


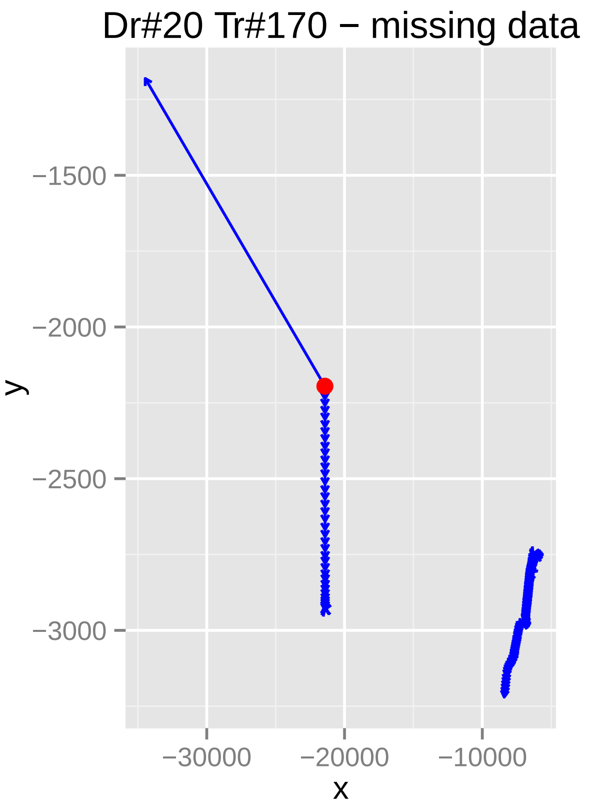
Dr#1 Tr#83 − missing data

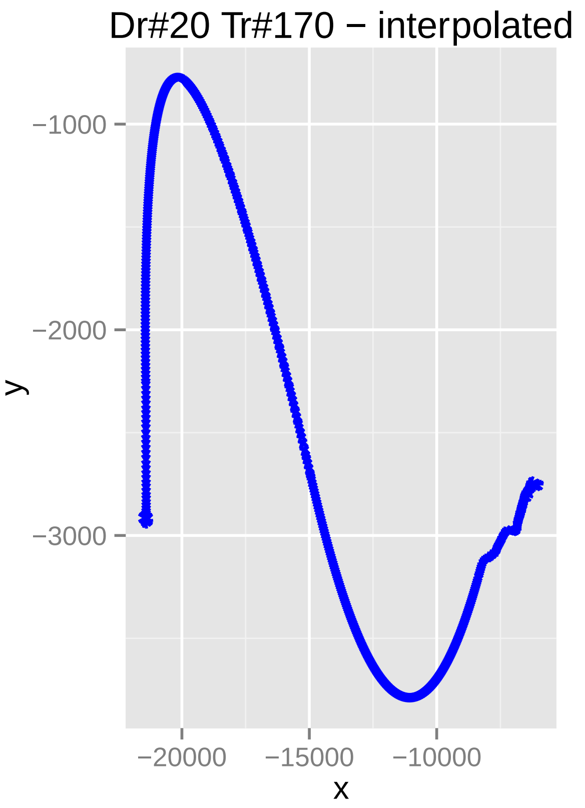


Dr#1 Tr#83 − interpolated

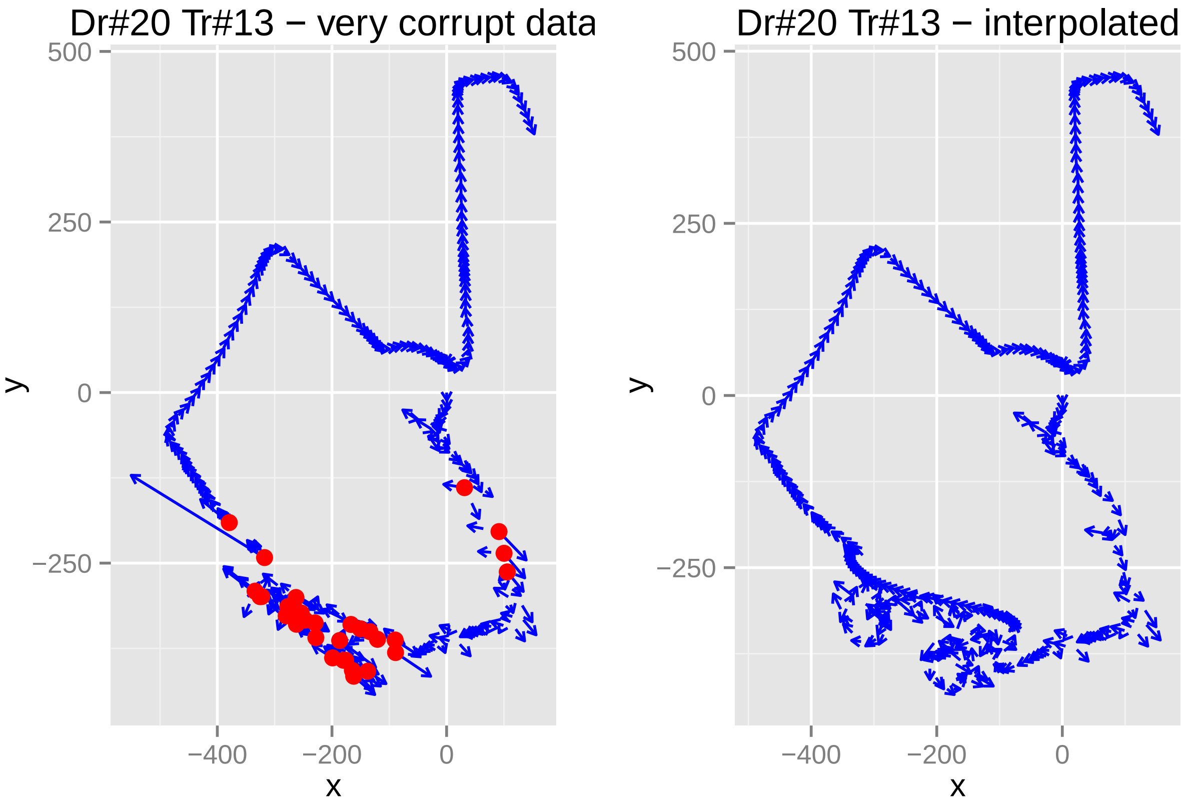








However, there are also many cases with data so corrupt that they cannot be reliably recovered:



We hence decided to limit recovery of missing data to cases with three or fewer missing sections, and discard the more seriously impaired cases. Overall, we recovered missing data for nearly 33,000 out of the over 40,000 affected driving trip data sets.

Additionally, we removed rows of data with:

- velocities $v_t < 2$ meters/second (about 4-5 miles/hour), because below that threshold cars are not meaningfully moving; and/or
- absolute angular velocities $|\Delta\theta_t| > 150$ degrees, because such turns are too sharp for cars to reasonably perform in one second (unless Hollywood action-movie cascadeurs happen to buy insurance from AXA. . .).

In terms of time cost, our various data verification and cleaning steps took over 100 hours on six CPU cores of a single computer.

2. Driver Identification as Classification Problem

Following the above data verification and cleaning procedures, the question for us to address now is that, with such a database of labeled personal driving trip data (raw GPS plus derived features), whether we can effectively distinguish among different drivers' different driving styles.

2.1. General Problem Framing

For each individual Driver D, we have got a collection of labeled driving data representing his/her typical driving habits. Because we have over 2,700 such drivers in the database, for each Driver D we also have an abundance of labeled driving data that are ***not*** by Driver D.

With such labeled data and a "one-vs.-all" approach, we can train a discriminative classification model to distinguish the driving style of Driver D versus an "average" driving style among others.

2.2. Classification Models' Granularity Level: Second-by-Second

A key modeling decision to make is at what level of granularity we should keep the data features. One possible, and computationally beneficial, choice is to reduce dimensionality by summarizing the features, i.e. velocity, acceleration, jerk, angular velocity and angular acceleration, at the trip level - that is, condensing each driving trip data set with hundreds or thousands of second-by-second observations to a single vector of averages, absolute values, maxima, minima, etc., of the features. However, such an approach would both lose and distort a great deal of information:

- Firstly, it would lose information on the mutual co-occurences of various value ranges of the features during a driving trip; e.g., when looking at trip-level summary statistics, it would be difficult for us to know whether a driver tends to take sharp turns at high velocities or whether he/she tends to speed when driving straight (when angular velocity is near zero).

- Secondly, common sense suggests that the observations that hold strong signals about individual driving styles are likely to be a small portion among the total recorded data on a typical driving trip; that is to say, most of the time most people drive very similarly - e.g. during generally slow urban street driving - and personal driving styles only manifest clearly in very specific maneuvers such as turning at considerable speeds or driving along a highway. If we summarize the features at the trip level, such valuable signals would be swamped by the majority data portions that are indiscriminative.

Because of the above reasons, we decided to build classification models at the ***granularity level of each second of each driving trip*** as follows: for each second t of observed data features

- velocity v_t,
- signed and absolute acceleration a_t and $|a_t|$,
- signed and absolute jerk j_t and $|j_t|$,
- signed and absolute angular velocity $\Delta\theta_t$ and $|\Delta\theta_t|$, and
- signed and absolute angular acceleration $\Delta\Delta\theta_t$ and $|\Delta\Delta\theta_t|$,

we asked if that combined observation at time t is more likely to have been generated by the subject driver or by another "average" driver. Note the ***implicit simplifying assumption of independence among different time instances*** during a driving trip. This is a strong assumption because there are surely non-zero correlations in practice, especially among consecutive instances. Nonetheless, this would turn out *not* to hamper the effectiveness of our approach.

A one-vs.-all classification model would be trained for each individual driver D on record, with a training data set comprising of 60% of driver D's clean/cleaned trip data sets and an equivalent number of trip data sets randomly sampled from other drivers. Each training data set typically has well over 100,000 labeled observations. At test time, the trained discriminative model would be given an unlabeled driving trip data set of length T seconds comprising GPS coordinate readings and the related derived higher-order features per second t, and the model would be asked if this trip is more likely to be by driver D or another driver. The prediction is performed as follows:

- First of all, the model would score each second t of the trip to produce the log of the odds that the observed features at that second t are by driver D;

- Then, a trip-level log-odds is calculated as the sum of the individual per-second log-odds from $t = 1$ to $t = T$, and the trip-level log-odds is compared with a certain decision threshold log-odds (zero by default, corresponding to a probability decision threshold of 50%) to produce the prediction.

2.3. Modeling Method: Random Forest

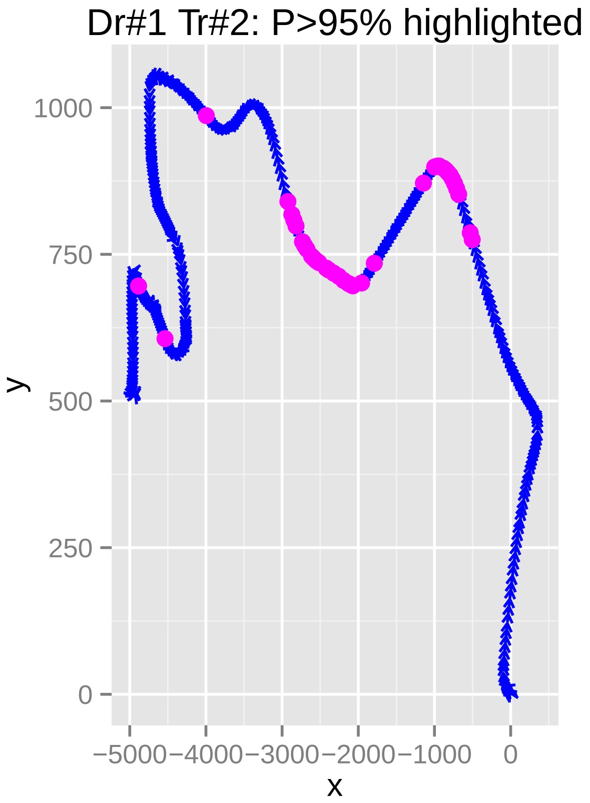
We opted for Random Forest for building the one-vs.-all classification models, because of this method's two key benefits:

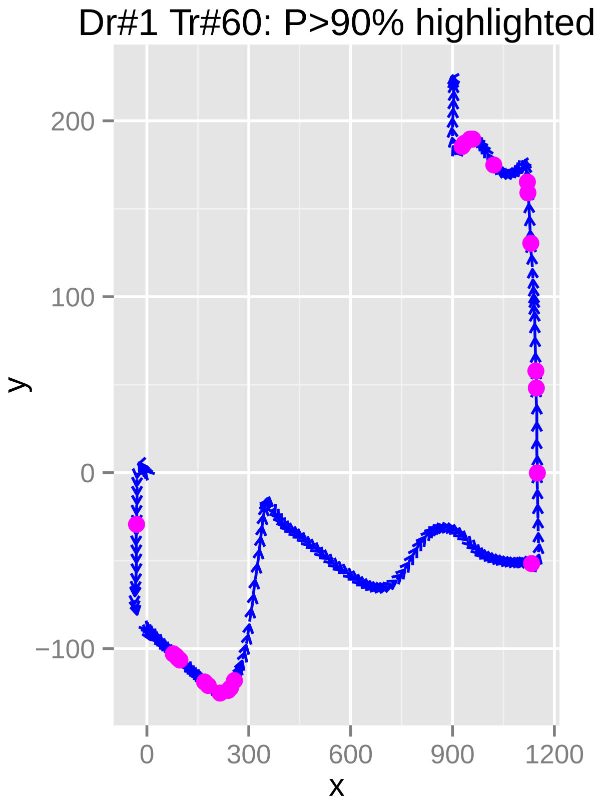
- Random Forest is fast to train because each individual classification tree is simple, and different trees can be trained in parallel; here, each of our Random Forest models consists of 480 trees trained in parallel on six CPU cores; and

- Random Forest automatically discovers highly relevant interactions among data features, which is crucial in our modeling tasks because individual driving styles are likely to be non-trivial interactions among velocity, acceleration, jerk, angular velocity and angular acceleration.

3. Results and Evaluation

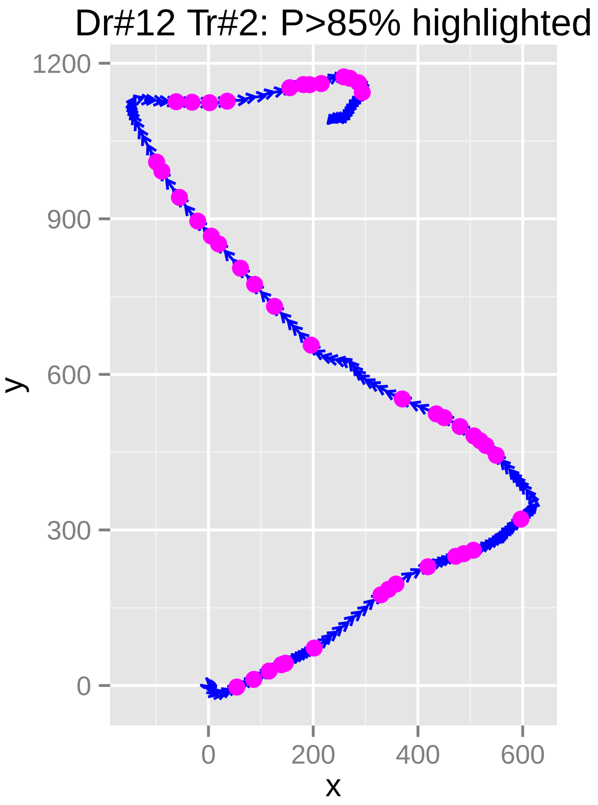
To illustrate our modelling approach, let us consider a one-vs.-all classification model trained on trips by an individual driver - Driver #1 - versus randomly sampled trips from other drivers. We can see what the model believes signifies Driver #1's driving style by looking at time instances for which the model predicts high positive-class probabilities:



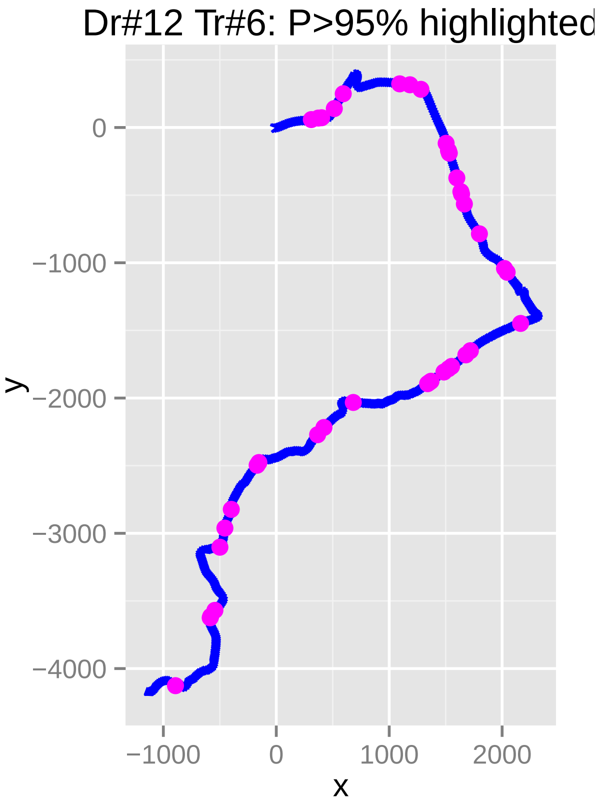


We can see that Driver #1's most characteristic maneuvers seem to cluster around bends, which suggests that his/her combinations of velocity, acceleration, jerk, angular velocity and angular acceleration just before, during and just after turning are very different from those by another "average" driver.

The signature styles differ quite diversely among drivers. For example, Driver #12 seem to have highly characteristic behaviors along straight roads or highways:



Dr#12 Tr#2: P>85% highlighted

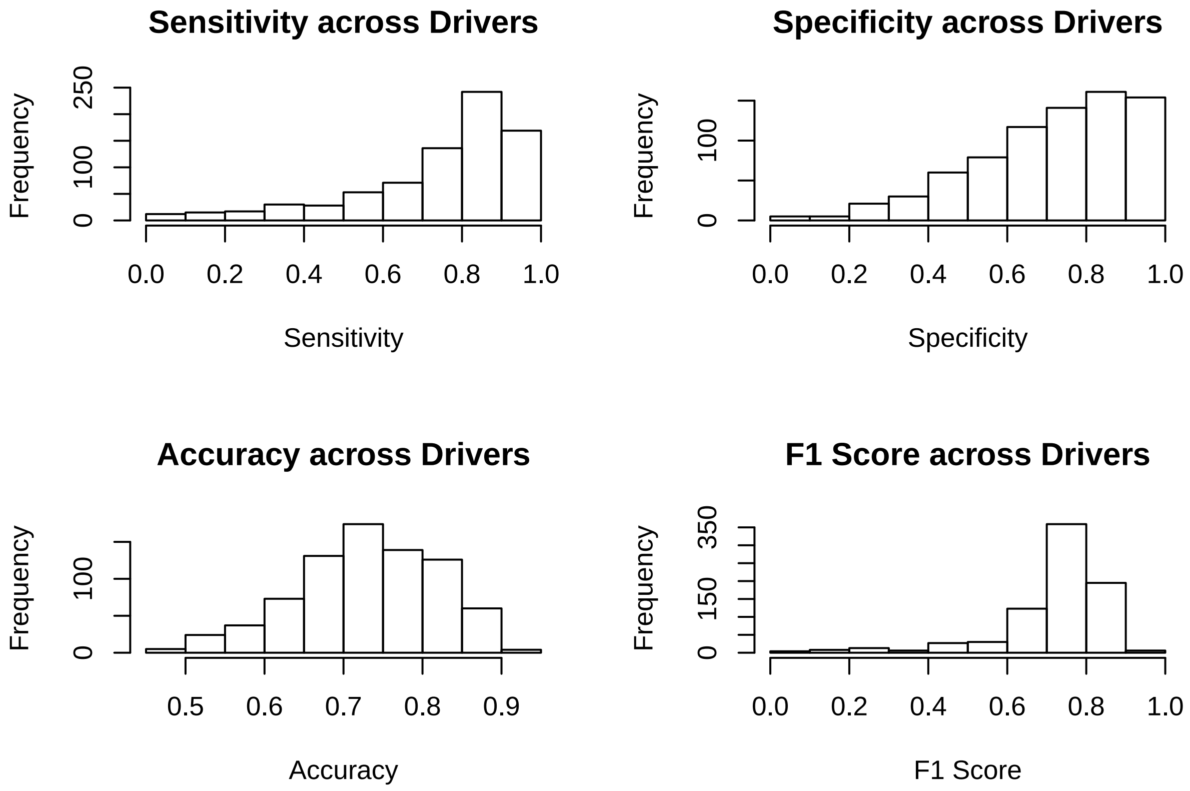


Dr#12 Tr#6: P>95% highlighted

Within the project's time and computing capacity constraints, we managed to train **773** one-vs.-all Random Forest models for 773 individual drivers in the database. At the default decision threshold log-odds of zero, our models have the following out-of-sample performance metrics on test cases comprising the 40% driving trip data sets not used in training:

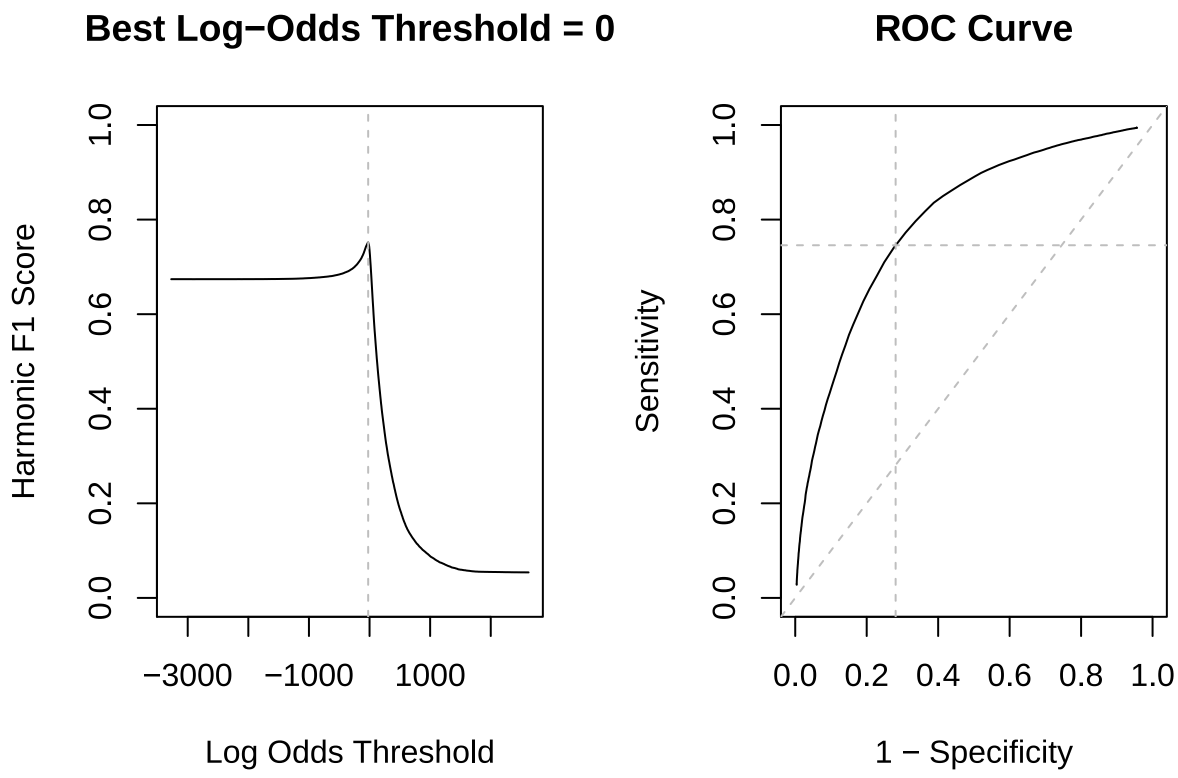
- **Combined Sensitivity**: 74.6%;
- **Combined Specificity**: 71.9%;
- **Combined Accuracy**: 73.2%; and
- **Combined Harmonic F1 Score**: 73.6%.

At this default threshold, the distributions of these metrics across individual drivers' test samples are as follows:



We can see that all of these metrics look very decent, concentrating on the high side of the (0, 1) spectrum, peaking in the 70-80% range.

It turns out that the default decision threshold is also approximately optimal in terms of maximizing the combined Harmonic F1 Score, achieving a highly favourable trade-off on the ROC curve:



4. Conclusion and Potential Directions for Improvement

In this project, we have demonstrated the efficacy in characterizing individual driving styles of features such as velocity, acceleration, jerk, angular velocity and angular acceleration derived from relatively high-frequency second-by-second GPS coordinate readings from telematics devices. The approach is simple in terms of both the small handful features used and the implicit assumption of independence among time instances during a driving trip. That this simple approach can achieve average accuracy of about 80% is very promising indeed.

Another reason to be optimistic about these results is that the labels provided by AXA are in fact noisy: of the 200 trips labeled with each driver D, there are an undisclosed minority number of random trips that are actually not by D. This is the case for anonymization purposes. Had the labels been entirely clean, as should be the case in a well-maintained corporate database, our models would have achieved even higher accuracy, perhaps in the 90%s.

Possibilities for improvement include:

- **Time Series Approach with Autocorrelations**: We expect accuracy gains when we relax the strong independence assumption and model the correlations among observations at different time instances.
- **Trip Pattern Matching**: we have so far ignored information such as the $(x_t,\, y_t)$ coordinates as well as total trip duration. Such information can potentially be used to detect personal itineraries that each driver frequently perform, such as work commute, school drop-off/pick-up, and shopping, and should contain a lot of signals about the identity of the driver. This approach, however, can be very contentious on privacy grounds.