A supplemental material for "Improving automobile insurance claims frequency prediction with telematics car driving data"

The online supplementary materials contain:

A Telematics data cleaning

B Limited fluctuation credibility model revisited

C `Keras` code for 1D CNN

A Telematics data cleaning

Telematics car driving data have a large size and are deemed big data. Data cleaning for such big data is very challenging since the same procedure needs to be applied to all the trips of all cars. Thus, the data cleaning should be adequately flexible for this aim.

We firstly visualize several typical trips to demonstrate what data issues need to be addressed during the cleaning procedure. Then, a "naive" data cleaning procedure is designed, and its performance is monitored on the selected trips. Finally, a "universally" applied data cleaning procedure is further derived.

A.1 Original telematics data

Three trips of three drivers are illustrated in the following figures. For each trip, 6 plots are presented:

1. Top-left: Time series of GPS signal quality, instrument panel signal quality, and accelerometer signal quality;
2. Top-right: Trajectory (x, y);
3. Middle-left: Time series of GPS speed $v^{(gps)}$, instrument panel speed (VSS speed) $v^{(vss)}$,

4. Middle-right: Time series of GPS heading $\psi^{(gps)}$;

5. Bottom-left: Time series of longitudinal acceleration $a^{(acc)}$;

6. Bottom-right: Time series of lateral acceleration $a'^{(acc)}$.

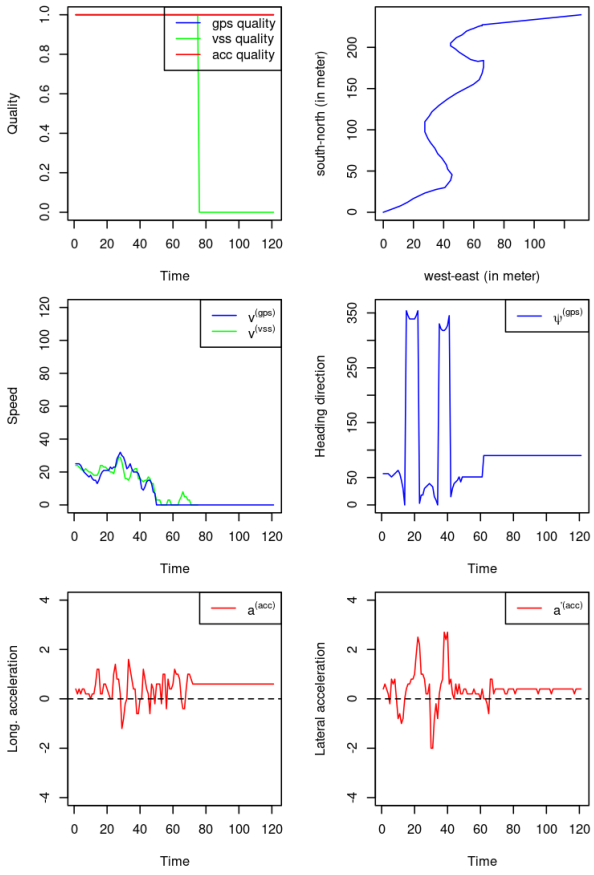

Figure 1: One trip of driver 8.

Figure 1 shows a very short trip of 2 minutes for driver 8. We have the following observations:

- The top-left plot shows that the instrument panel signal is missing in the last 40 seconds.

- The top-right and middle-right plots show that the vehicle starts from east-south and approaches to west-north. Note that there are jumps between $\psi^{(gps)} = 0$ and $\psi^{(gps)} = 360$.

- The middle-left plot demonstrates that GPS speeds match with VSS speed, and VSS speeds are missing for the last 40 seconds.

- The bottom two plots reveal that there are **calibration bias** with both the acceleration rates, which might be corrected by subtracting the median of acceleration rates.

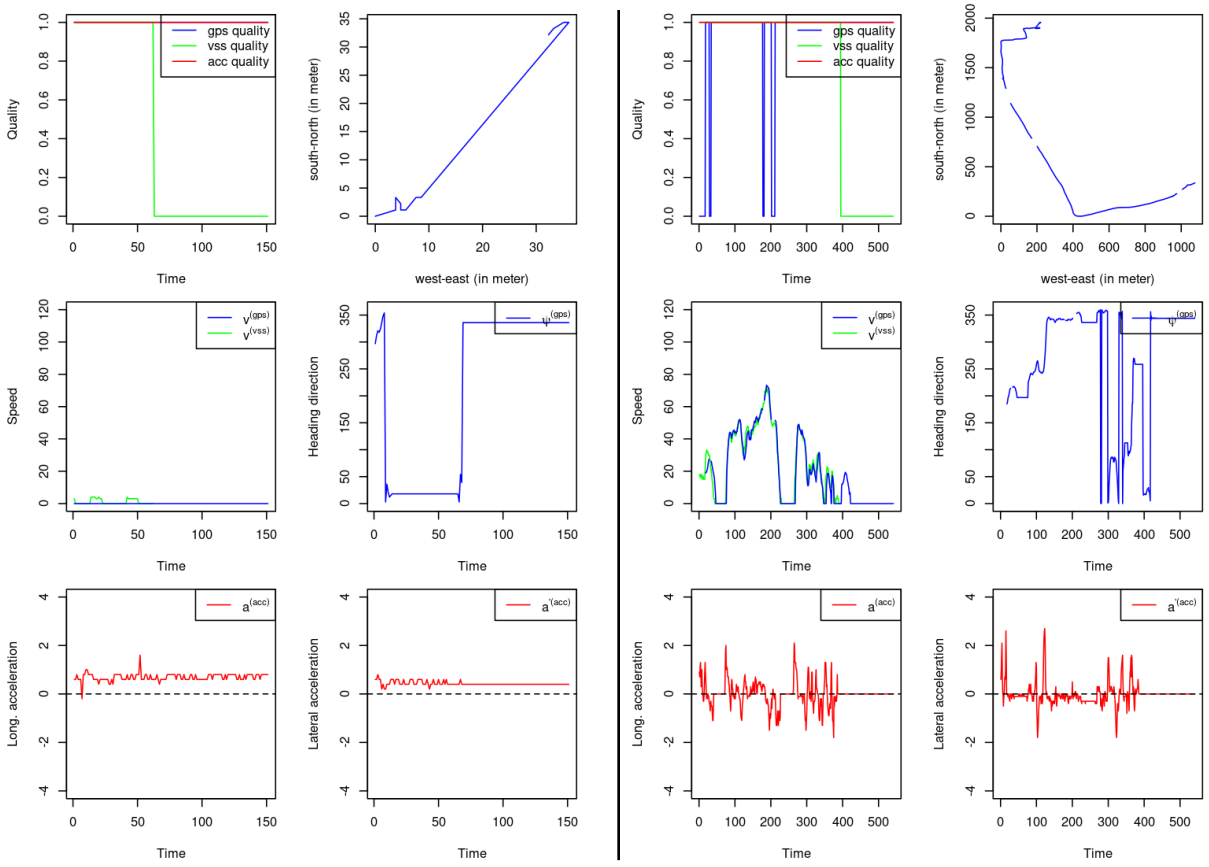

Figure 2: Two trips of driver 8.

Figure 2 shows another two trips of driver 8. We have the following observations:

- For the left trip, the bottom two plots capture the calibration bias of accelerometer again.

- For the right trip, there are several segments with a missing GPS signal. We need to interpolate the GPS coordinates (x, y), speed $v^{(gps)}$ and heading $\psi^{(gps)}$ when the GPS signal is missing.

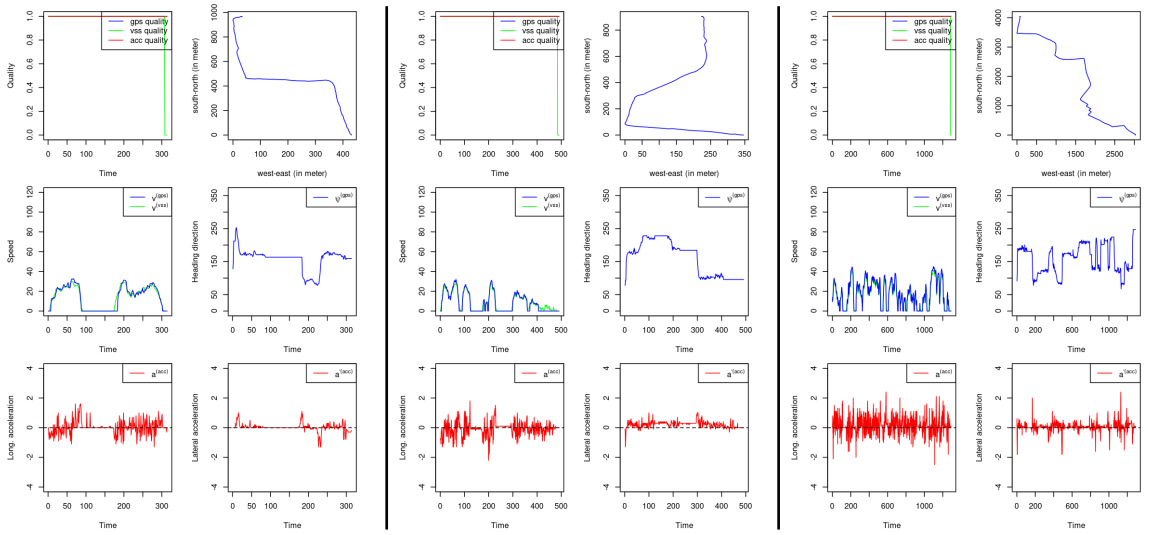

Figure 3: Three trips of driver 288.

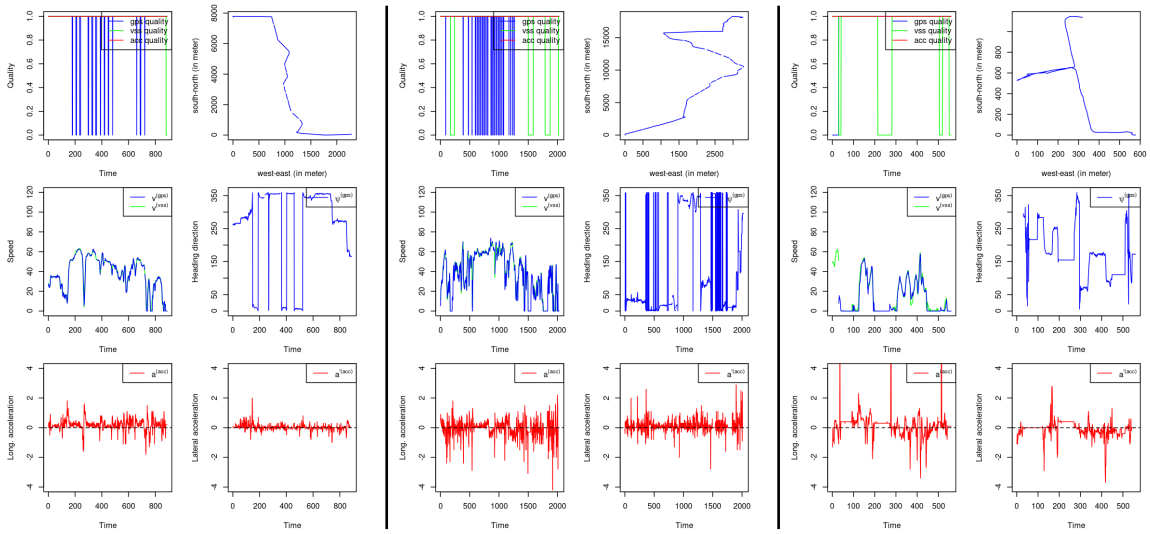

Figure 4: Three trips of driver 1188.

Figure 3 displays three trips of driver 288. GPS signals seem stable for the three trips, and the accelerometer seems work well except for the lateral acceleration during the second trip. Figure 4 presents three trips of driver 1188. GPS signals are unstable in the first two trips, and instrument panel signals are unstable for the last two trips. There seems to be calibration bias of acceleration rates for all these trips.

In summary, we need to consider the following data cleaning issues:

- There are missing values in GPS coordinates (x, y), GPS speed $v^{(gps)}$, GPS heading

$\psi^{(gps)}$, and VSS speed $v^{(vss)}$. These missing values need to be interpolated.

- There are frequent calibration issues with the accelerometer variables $a^{(acc)}, a'^{(acc)}$. It is difficult to remove the calibration bias since the timing and severity of such a bias are rather random. Consequently, it is better to consider other variables to describe the acceleration in two directions. We will use the derived longitudinal acceleration rates $a^{(gps)}, a^{(vss)}, a^{(xy)}$ which are obtained from GPS speed $v^{(gps)}$, VSS speed $v^{(vss)}$, and GPS coordinates (x, y), respectively. And we will replace lateral acceleration $a'^{(acc)}$ by angle changes of heading directions $\Delta^{(gps)}, \Delta^{(xy)}$, which are derived from GPS heading $\psi^{(gps)}$ and GPS coordinates (x, y), respectively.

A.2 Selection of telematics variables

After data imputation and derivation of the corresponding telematics, we discuss the telematics variable selection among speed $v^{(gps)}, v^{(vss)}, v^{(xy)}$, acceleration $a^{(gps)}, a^{(vss)}, a^{(xy)}$, angle $\psi^{(gps)}, \psi^{(xy)}$, and angle change $\Delta^{(gps)}, \Delta^{(xy)}$. Those telematics variables are plotted in Figure 5 for the same trips as those in Figures 1, 2, 3 and 4. We have the following observations for Figure 5:

- The top two plots in Figure 5 are exactly the same as the top two in Figure 1.
- In the middle-left plot, we add $v^{(xy)}$ for comparison. Note that there is a GPS drift around 60 second causing a jump of $v^{(xy)}$.
- In the middle-right plot, we add $\psi^{(xy)}$ for comparison. The derived heading direction $\psi^{(xy)}$ is always zero when the vehicle stands still after 50 seconds.
- In the bottom plots we show the derived acceleration $a^{(gps)}, a^{(vss)}, a^{(xy)}$ and the derived angle change $\Delta^{(gps)}, \Delta^{(xy)}$.

We now investigate the GPS drift at around the 60th second. In Figure 6, time series of (x, y) coordinates, $v^{(xy)}$ and $\psi^{(xy)}$ from the 45th second to 65th second are plotted, respectively. A jump of (x, y) can be seen at the 62nd second, which leads to an extreme speed jump from 0 to more than 200 km/h and an unusual direction jump from 0 to more than 60 degree in one second.

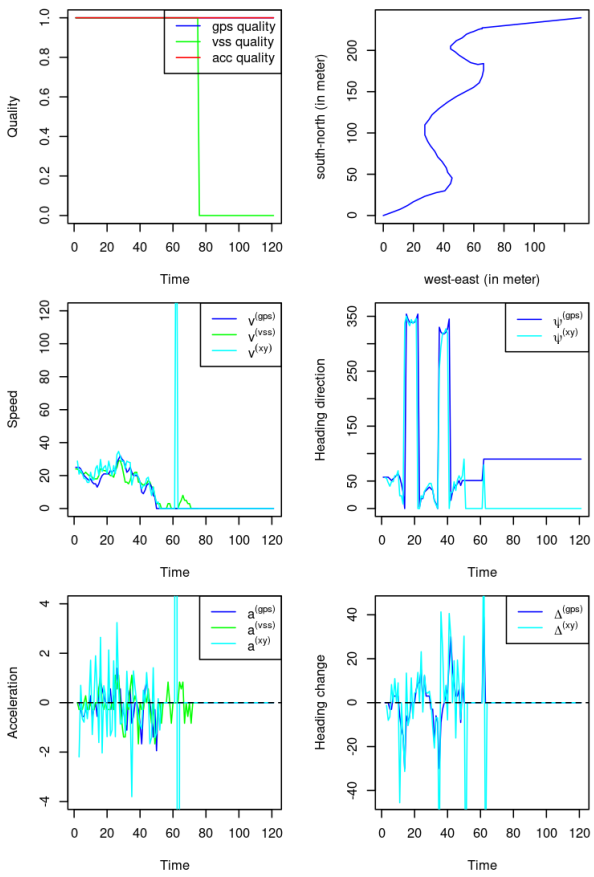

Figure 5: One trip of driver 8.

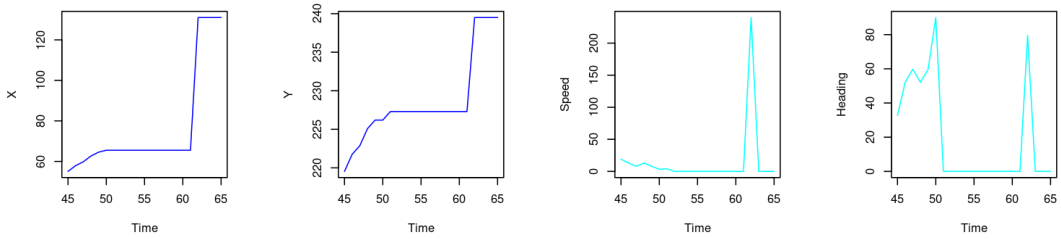

Figure 6: GPS drift.

We conclude that the derived variables using GPS coordinates (x, y) are unstable compared with those using GPS speed, GPS heading and VSS speed. This is due to the measurement error of GPS coordinates and its leverage effects on the acceleration and angle change.

We have the following observations for Figure 7:

- For the second trip of driver 8, we see a GPS drift at around the 70th second.

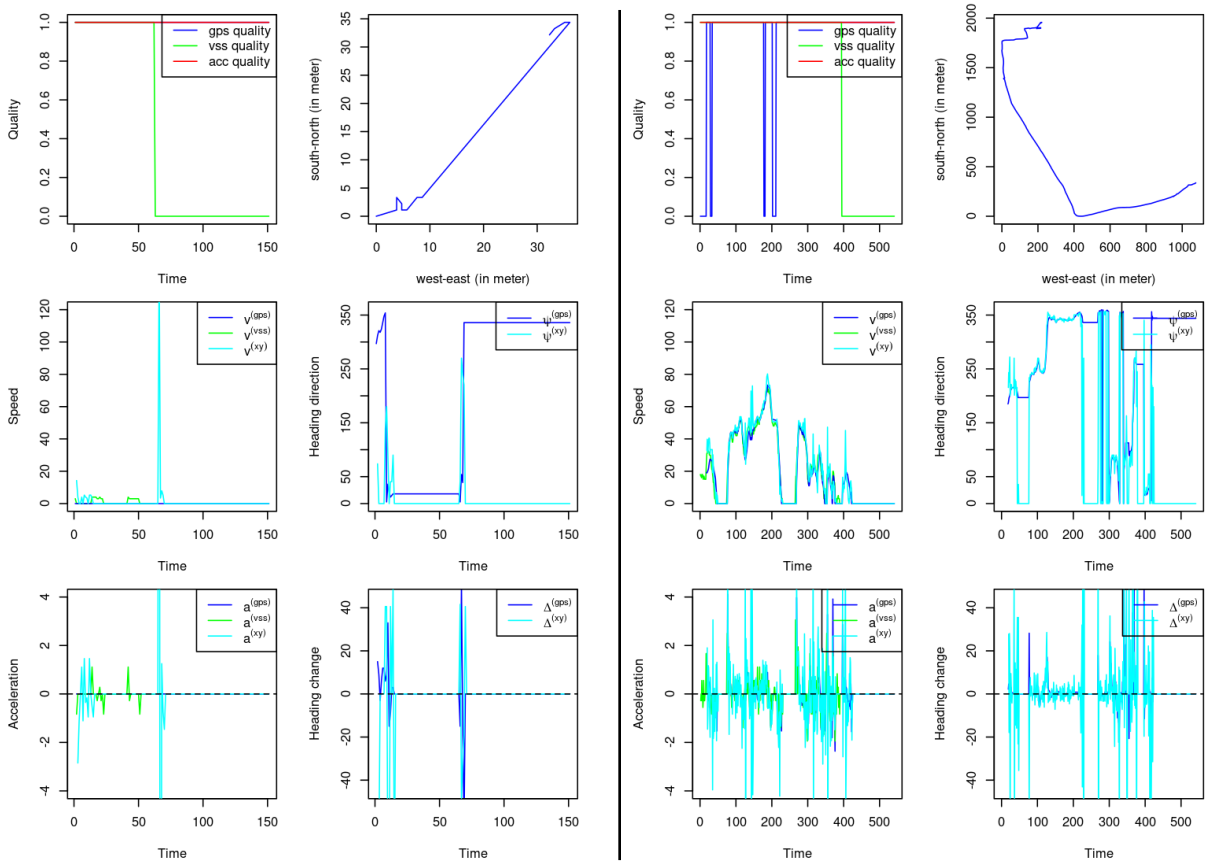

Figure 7: Two trips of driver 8.

- For the third trip of driver 8, the imputation works very well for GPS coordinates, GPS speed and GPS heading. GPS drifts are often observed when there is a speed peak. The heading $\psi^{(xy)}$ derived from GPS coordinates is always incorrectly zero when the vehicle stops. Again, the acceleration and angle change derived from GPS coordinates are very unstable.

There are no new observations for Figures 8 and 9.
In summary, we argue that

- The linearly imputation works reasonably well.

- Due to GPS coordinates drift, we should avoid using variables derived from (x, y). Instead, the telematics variables $v^{(vss)}, v^{(gps)}, a^{(vss)}, a^{(gps)}, \psi^{(gps)}, \Delta^{(gps)}$ are more reliable.

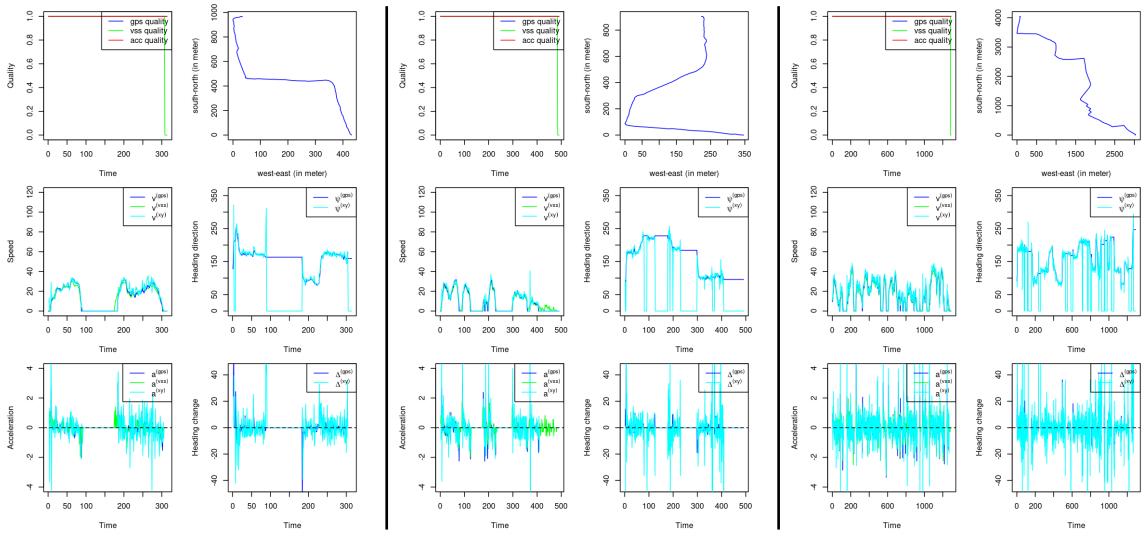

Figure 8: Three trips of driver 288.

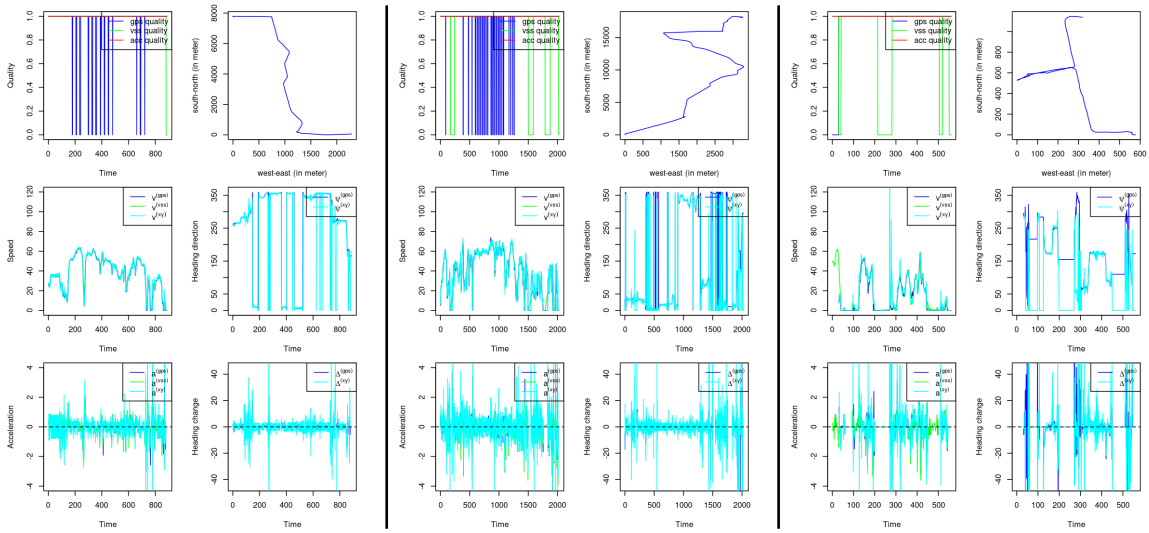

Figure 9: Three trips of driver 1188.

- The distance should be derived using the speed variable rather than GPS coordinates (x, y).
- The three variables v, a, Δ are related to driving behavior, while heading direction $\psi^{(gps)}$ is irrelevant.
- Variables from the same sensor should be adopted for a certain study, i.e., we should use either $(v^{(vss)}, a^{(vss)})$ or $(v^{(gps)}, a^{(gps)}, \Delta^{(gps)})$ only.

We plot speed $v^{(gps)}$, heading direction $\psi^{(gps)}$, acceleration $a^{(gps)}$, angle change $\Delta^{(gps)}$ for the previously investigated trips of drivers 288 and 1188 in Figures 10 to 11. The timing of missing values is denoted by a horizontal line at the top of each plot. Note that we have capped the $a^{(gps)}$ between $(-4, 4)\text{m/s}^2$, and $\Delta^{(gps)}$ between $(-45°, 45°)$. Finally, the missing $v^{(gps)}$ and $\psi^{(gps)}$ are interpolated linearly, and $a^{(gps)}$ and $\Delta^{(gps)}$ are obtained from the imputed values. Those telematics variables are shown in Figures 12 to 13, which indicates that we have done an appropriate data cleaning.

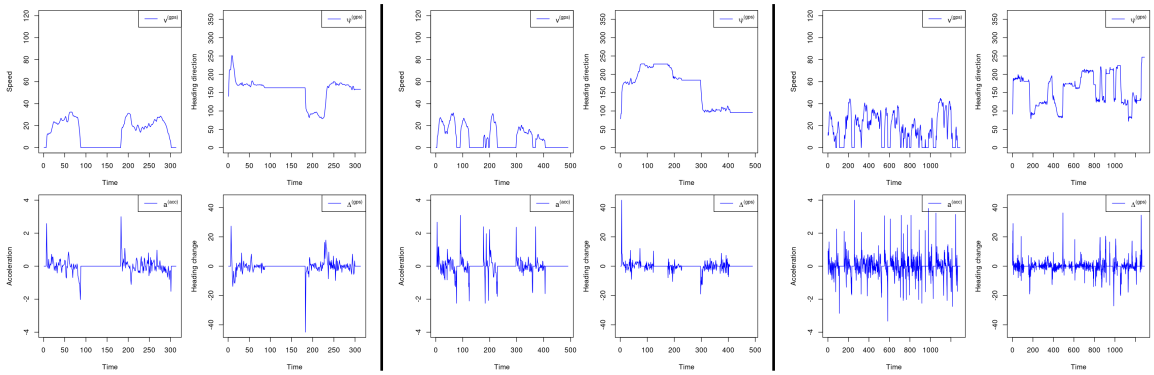

Figure 10: Three trips of driver 288.

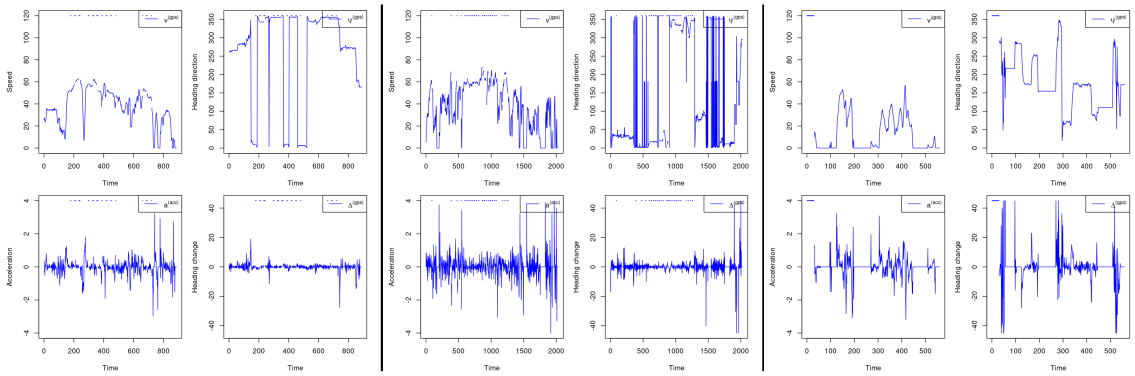

Figure 11: Three trips of driver 1188.

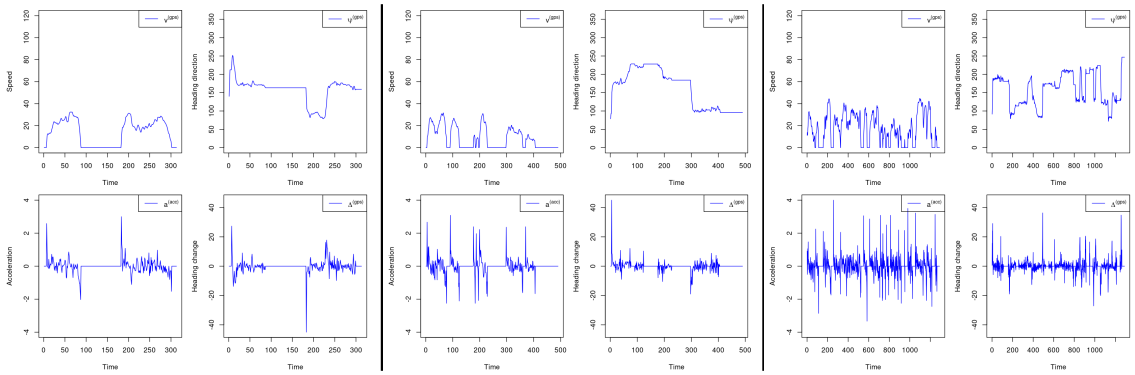

Figure 12: Three trips of driver 288.

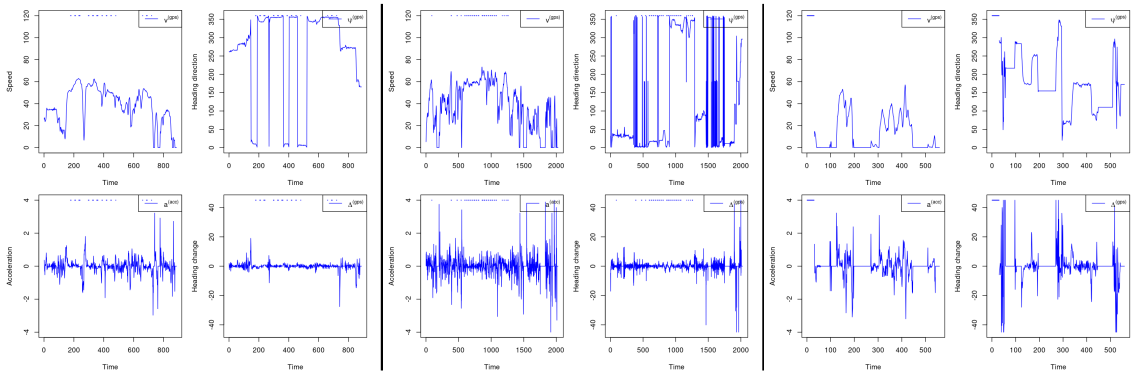

Figure 13: Three trips of driver 1188.

B Limited fluctuation credibility model revisited

According to the Central Limit Theorem, average risk score $\bar{\phi}_i$ approximately follows a Gaussian distribution with mean μ_i and variance σ_i^2/c_i

$$\bar{\phi}_i \ \dot{\sim} \ \mathcal{N}(\mu_i, \sigma_i^2/c_i),$$

where

$$\bar{\phi}_i = \frac{1}{c_i}\sum_{j=1}^{c_i} \hat{\phi}(z_{i,j})$$

for large number of trips c_i. To fully credit an average risk score, it needs to be close to μ_i sufficiently with a high probability. In other words, the minimal number of trips is determined such that the following inequality holds:

$$\Pr(\mu_i - r\mu_i \leq \bar{\phi}_i \leq \mu_i + r\mu_i) \geq 1 - \alpha. \tag{1}$$

Inequality (1) defines a limited fluctuation credibility model, which can be solved as follows:

$$\begin{aligned} \Pr(\mu_i - r\mu_i \leq \bar{\phi}_i \leq \mu_i + r\mu_i) &\geq 1 - \alpha \\ \Pr\left(-\frac{r\mu_i}{\sigma_i/\sqrt{c_i}} \leq \frac{\bar{\phi}_i - \mu_i}{\sigma_i/\sqrt{c_i}} \leq -\frac{r\mu_i}{\sigma_i/\sqrt{c_i}}\right) &\geq 1 - \alpha \\ \frac{r\mu_i}{\sigma_i/\sqrt{c_i}} &\geq \Phi^{-1}(1-\alpha/2) \\ c_i &\geq \left(\frac{\Phi^{-1}(1-\alpha/2)}{r}\right)^2 \left(\frac{\sigma_i}{\mu_i}\right)^2 \end{aligned} \tag{2}$$

where $\Phi^{-1}(1-\alpha/2)$ is the $(1-\alpha/2)$ percentile of standard normal distribution. We call

$$c_i^{(f)}(\alpha, r) = \left(\frac{\Phi^{-1}(1-\alpha/2)}{r}\right)^2 \left(\frac{\sigma_i}{\mu_i}\right)^2 \tag{3}$$

as the standard of full credibility for an average risk score $\bar{\phi}_i$. Let $\alpha = 10\%, r = 10\%$, we have

$$c_i^{(f)}(\alpha = 10\%, r = 10\%) = 271 \times \left(\frac{\sigma_i}{\mu_i}\right)^2.$$

If the number of individual trips does not satisfy the standard of full credibility (3) (i.e., $c_i < c_i^f$), we can use the credibility average risk score, defined as follows

$$\tilde{\phi}_i = Z_i\bar{\phi}_i + (1 - Z_i)\mu,$$

where Z_i is partial credibility and μ is the overall expectation of individual trip risk score. There are a variety of approaches developed for partial credibility, which are usually justified on intuitive rather than theoretical grounds. One of those justifications is to control the variance of credibility average risk score, such that

$$\text{Var}(\tilde{\phi}_i) = \text{Var}\left(\frac{1}{c_i^{(f)}} \sum_{j=1}^{c_i^{(f)}} \hat{\phi}(z_{i,j})\right). \tag{4}$$

Following

$$\text{Var}(\tilde{\phi}_i) = Z_i^2 \sigma_i^2 / c_i$$

and

$$\text{Var}\left(\frac{1}{c_i^{(f)}} \sum_{j=1}^{c_i^{(f)}} \hat{\phi}(z_{i,j})\right) = \sigma_i^2 / c_i^{(f)},$$

equation (4) leads to the following formula for partial credibility

$$Z_i = \min\left(1, \sqrt{\frac{c_i}{c_i^{(f)}}}\right).$$

C `Keras` code for 1D CNN

Listing 1: `Keras` code for 1D CNN specification.

```
build_model_cnn <-
  function(q1, q2, q3, s1, s2, s3, rate1, rate2) {
    ### input layer
    trips <-
      layer_input(shape = c(300, 5),
                  dtype = "float32",
                  name = "trips")
    ### convolutional neural network
    trips_score = trips %>%
      layer_conv_1d(
        filters = q1,
        kernel_size = s1,
        activation = "tanh",
        name = "cov1"
```

```
    ) %>%
    layer_average_pooling_1d(pool_size = s2, name = "ave1") %>%
    layer_conv_1d(
      filters = q2,
      kernel_size = s3,
      activation = "tanh",
      name = "cov2"
    ) %>%
    layer_global_average_pooling_1d(name = "ave2") %>%
    layer_dropout(rate = rate1, name = "drop1") %>%
    layer_dense(units = q3,
                activation = "tanh",
                name = "dense1") %>%
    layer_dropout(rate = rate2, name = "drop2") %>%
    layer_dense(
      units = 1,
      activation = "sigmoid",
      weights = list(array(c(0), dim = c(q3, 1)), array(0, dim = c(1)))
          ,
      name = "dense2"
    )
  ### compile model
  model <- keras_model(inputs = trips, outputs = trips_score)
  model %>% compile(
    optimizer = optimizer_adam(),
    loss = "binary_crossentropy",
    metrics = c("accuracy")
  )
  model
}
model1 <-
  build_model_cnn(
    q1 = 32,
    q2 = 16,
    q3 = 8,
    s1 = 7,
    s2 = 5,
```

```
    s3 = 7,
    rate1 = 0.5,
    rate2 = 0.5
  )
```

Listing 2: `Keras` code for 1D CNN calibration.

```
ind_test <- 2
patience_cnn <- 20
epoch_cnn <- 1000
batch_cnn <- 512
early_stop <-
  callback_early_stopping(monitor = "val_loss", patience = patience_cnn)
check_point <-
  callback_model_checkpoint(
    paste("./CallBack/best_model_",
          ind_test,
          sep = ""),
    monitor = "val_loss",
    verbose = 0,
    save_best_only = TRUE,
    save_weights_only = TRUE
  )
history1 <- model1 %>% fit(
  trips_train,
  risks_train,
  batch_size = batch_cnn,
  epochs = epoch_cnn,
  validation_data = list(trips_valid, risks_valid),
  callbacks = list(early_stop, check_point)
)
model1 %>%
  load_model_weights_hdf5(paste("./CallBack/best_model_",
                                ind_test,
                                sep = ""))
test_score <- predict(model1, trips_test)
```