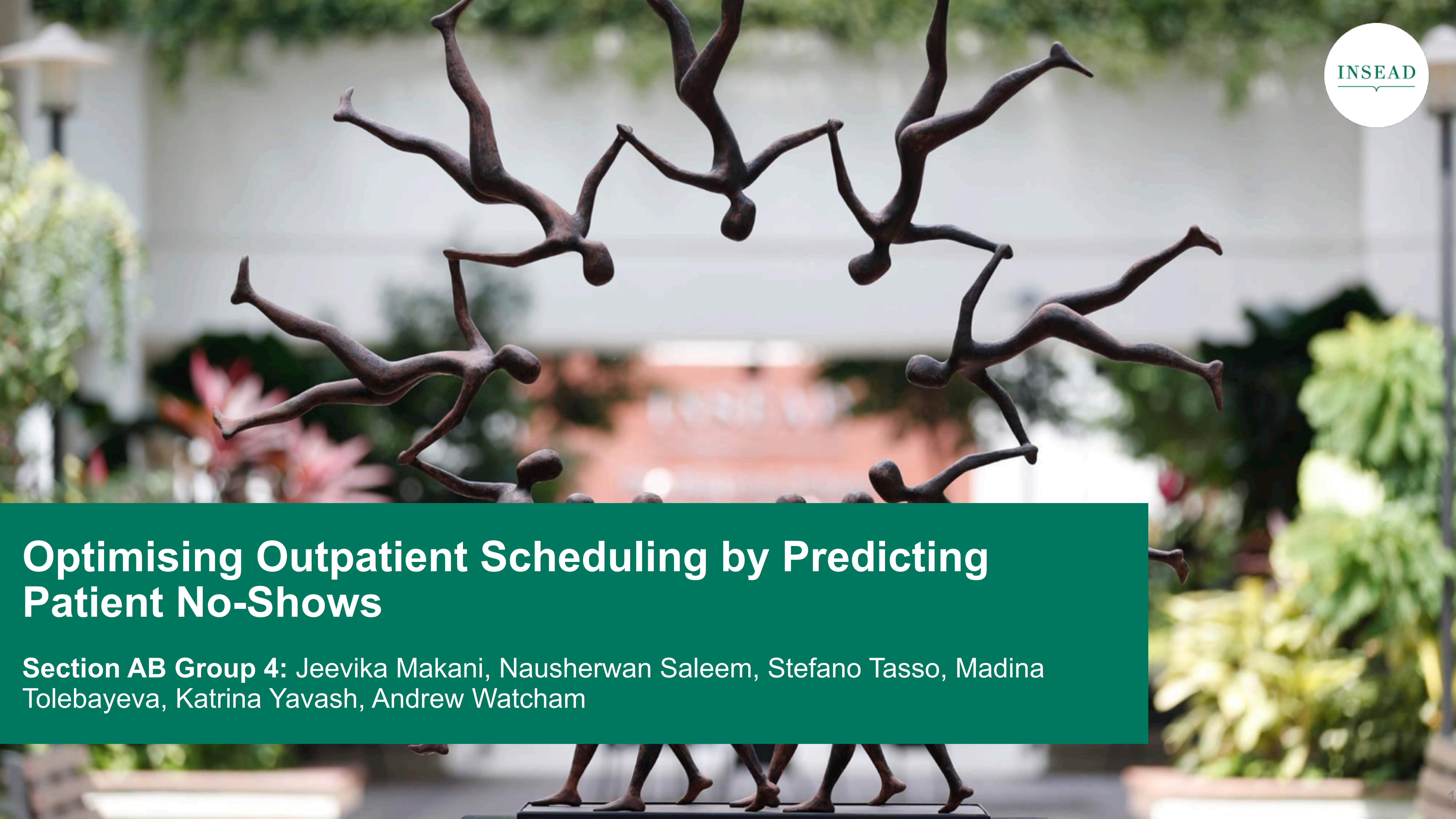


Optimising Outpatient Scheduling by Predicting Patient No-Shows

Section AB Group 4: Jeevika Makani, Nausherwan Saleem, Stefano Tasso, Madina Tolebayeva, Katrina Yavash, Andrew Watcham

Business Problem

INSEAD

- No-shows to doctors' appointments not only result in **lost revenue** for hospitals, but also longer-than-necessary **waiting lists** for patients

- If hospitals could more accurately predict which patients would not show up to the appointments, they would be able to generate greater profits by **overbooking their outpatient clinics** with minimal disruption to patients and medical staff

Data Description

- Outpatient appointments occurring over a 6-week period in 2016 for a hospital in the state of Espirito Santo in Brazil (110,527 appointments in the dataset)
- 13 original independent variables:
    - PatientID
    - AppointmentID
    - Gender (65% women)
    - ScheduledDay
    - AppointmentDay
    - Age (average age 37)
    - Neighbourhood (81 in total, of which 16 account for 50% of all appointments and 50% of no-shows)
    - Scholarship (enrolment in a government social welfare programme, true for 10%)
    - Hipertension (true for 20%)
    - Diabetes (true for 7%)
    - Alcoholism (true for 3%)
    - Handcap (2% have at least 1 handicap)
    - SMS_received (true for 32%)
- Prediction outcome: No-shows (20% in the original dataset)

Process Outline

- **Step 1:** Define the objective function
- **Step 2:** Inspect and clean the data
- **Step 3:** Feature engineering
- **Step 4:** Split the dataset into a training set (80% of the dataset), a validation set (10% of the dataset) and a testing set (10% of the dataset)
- **Step 5:** Run three prediction models (A) Logistic regression models, (B) RPART models, and (C) Random forest models
- **Step 6:** Choose the most appropriate model to predict no-shows based on the predicted profitability when applied to the test set

Step 1: Define the objective function

Hospital's profit will change depending on scenario:

- Book to capacity and there are no-shows
- Book to capacity and all patients attend their appointments
- Overbook and there are no-shows
- Overbook and all patients attend their appointments

With 21 appointment slots over a 7-hour day, the profit function in USD is:

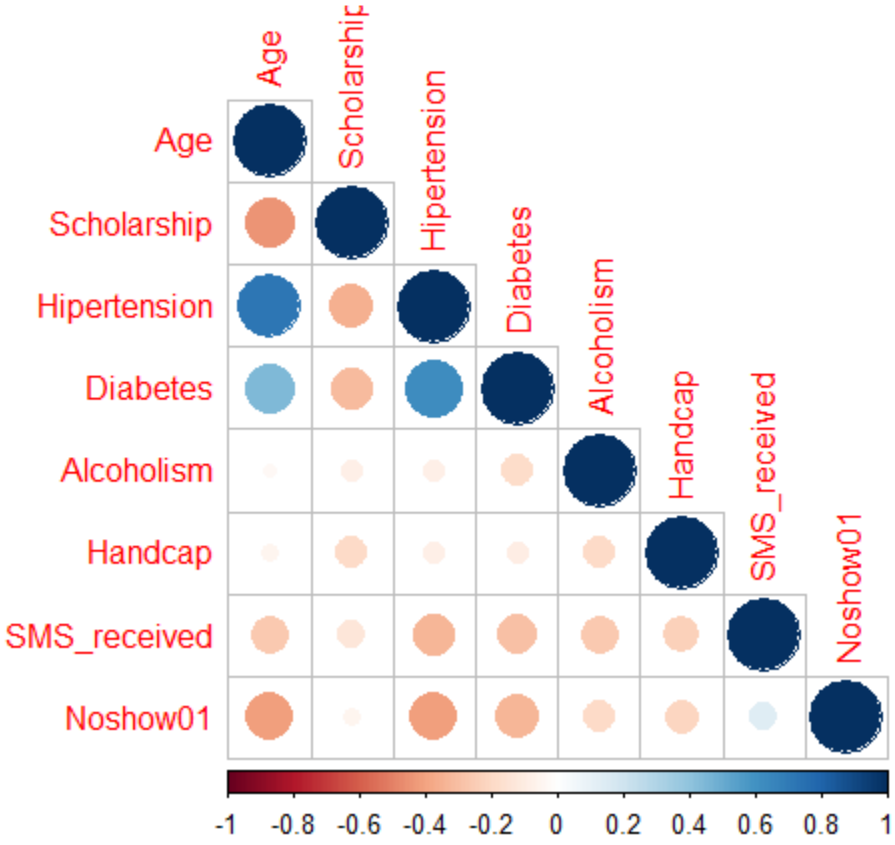
(47-2) * (B-N) - 21 * 33 - 66 * min{B-21-N,0}

- B is the number of appointments booked,
- N is the number of no shows
- In order to make a profit: 45 * (B-N) - 66 * min{B-21-N,0} must exceed 693

Estimates per a 20-min appointment: (i) Average revenue - $47, (ii) Fixed costs - $33, (iii) Variable costs - $2, (iv) Overtime costs if over capacity (2x FC) - $66

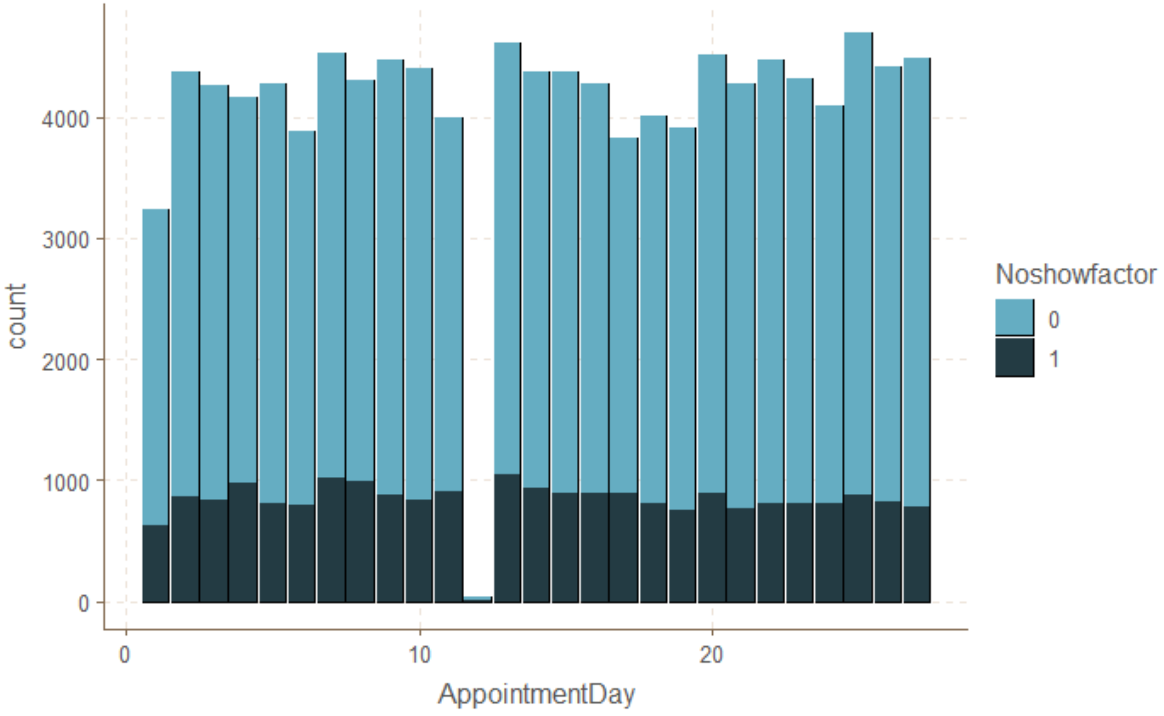
Step 2: Inspect and clean the data (1/3)

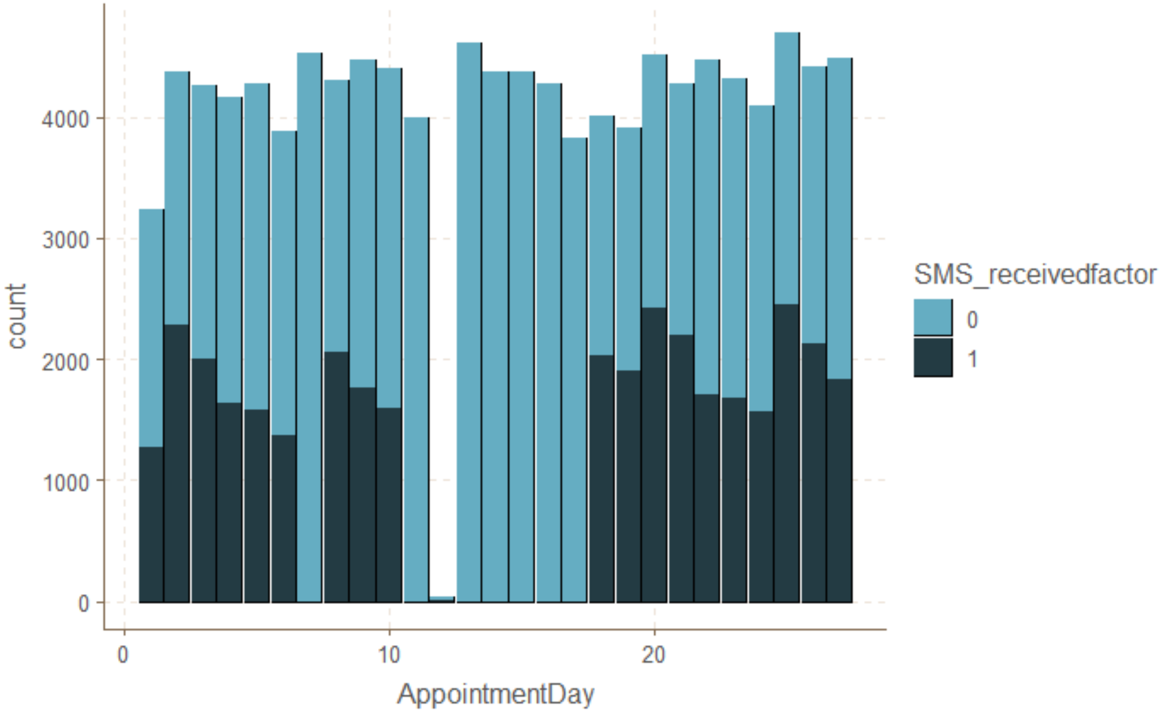
- Converted all quantifiable values into numeric data in order to assess correlations
- Highest correlations with the “no-shows”
  - Age (-0.42) – older patients are less likely to miss their appointments
  - Hypertension (-0.42) and Diabetes (-0.34) - patients with hypertension (high blood pressure) or diabetes are less likely to miss their appointments



Step 2: Inspect and clean the data (2/3)

- Proportion of patients who do not show up to their appointments has remained relatively stable (chart 1)
- Inconsistent pattern in SMS reminders (chart 2)
- Suggests SMS reminders do not materially alter patient behaviour





Step 2: Inspect and clean the data (3/3)

- Large dataset and few suspected errors so we opted to remove rows of data where likely erroneous values occurred

- Examples of inconsistencies include:

  - Patients whose **age changed by more than one year** over the sample period* (533 rows)

  - Patients who **appeared to change gender** over the sample period** (263 rows)

  - Appointments which were **scheduled after the date of the appointment** (5 rows)

  - Patients with **negative ages** (1 row)

** The sample period covered six weeks of appointments*
*** There were 263 instances of patients codified as male in one appointment and female in another. While some of these cases may not be erroneous, a transition rate of (263 instances divided by two to avoid double counting divided by 110,527) 0.12% of the population every six weeks translates into over 1% per annum, which exceeds a realistic rate of transgender transitioning*

Step 3: Feature engineering

- **Prior no-shows** variable showed number of prior no shows *(as a proxy for unreliability)*
- **"Same day"** variable (binary) for appointments scheduled on the same day that they occur *(no shows are less likely for such appointments)*
- Variable for the **time at which appointments were scheduled** *(earlier in the day an appointment is booked, the more organised the patient is)*
- **Handicap dummy variable** *(patients with handicaps more likely to have a system and support in place to reach their appointments)*
- **Lagged days dummy** - time between scheduling and the actual appointment *(greater this length of time, the more likely that a patient will not show up for their appointment)*
- **Age group categories** *(patients to take greater care of their health as they age)*
- Additionally we (i) dropped variables that could be replaced by new features; (ii) converted variables into the appropriate type of factor; (iii) dropped PatientID and AppointmentID to make predictions more generalisable

Step 4: Split the data into training, validation and final testing sets

We spit our data set such that:

80% of data went into the training set

10% of data went into the validation set (to be used to test model iterations)

10% of data went into the testing set

Step 5A: Run prediction models – Logistic Regression Model

Accuracy:
~92%

- Started with a model containing all variables

```
model_logistic_1<- glm(formula = Noshow01 ~ Gender + Neighbourhood + Scholarship + Hipertension +
Diabetes + Alcoholism + SMS_received + appointment_date + sameday + handicap_dummy + laggeddays +
age_group + appointment_day + priornoshows, family = binomial(link = "logit"),data = training_data)
```

- Applied this model to the validation set and calculated profit according to the profit function given the test predictions

```
predictiontable1
```

```
```

```
         actual
predicted    0    1
        0 8073  182
        1  744 2052
```

```
Profitfunction1 = 45*predictiontable1[1,1] + -21*predictiontable1[2,1] +
-36*predictiontable1[1,2] + 36*predictiontable1[2,2]
Profitfunction1
```

```
```

```
[1] 414981
```

- Produced new versions of the model iteratively to maximise the profit output from the validation set
- Finalised a logistic model that typically maximised the output from the profit function

```
predictiontable2test
```

```
```

```
         actual
predicted    0    1
        0 8040  121
        1  778 2113
```

```
Profitfunction2test = 45*predictiontable2test[1,1] +
-21*predictiontable2test[2,1] + -36*predictiontable2test[1,2]
+36*predictiontable2test[2,2]
Profitfunction2test
```

```
```

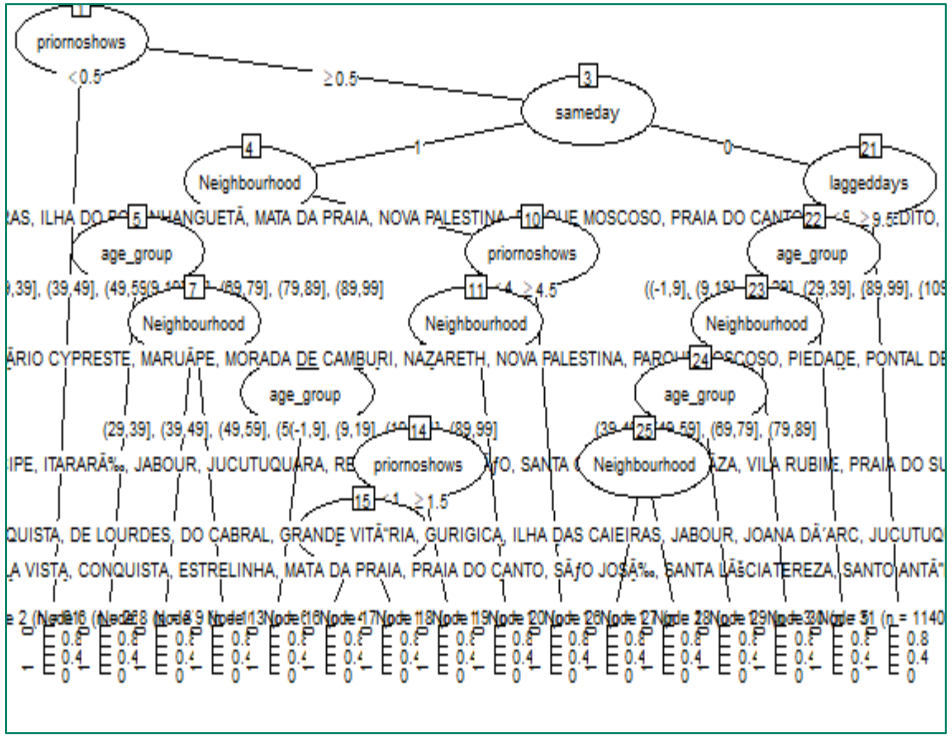
```
[1] 417174
```

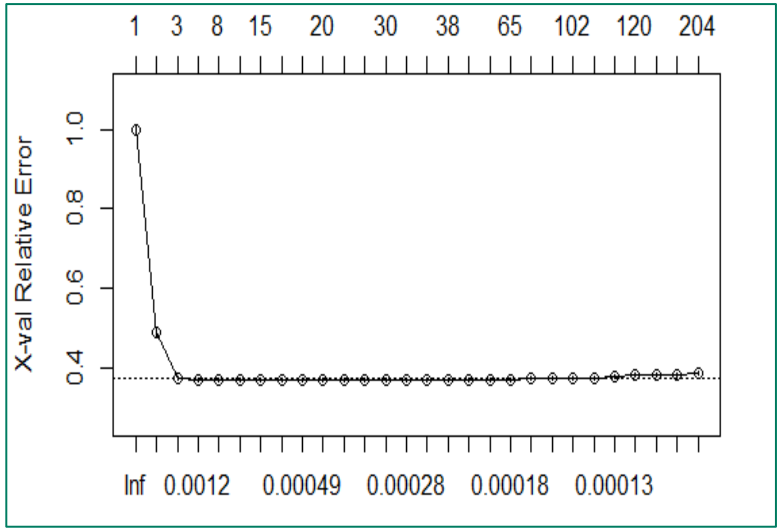
Step 5B: Run prediction models – RPART Model

Accuracy:
~92%

INSEAD

- Also experimented with RPART model:
  - Iteratively varied complexity parameter (CP) levels and variables to optimize the model
  - Evaluated effectiveness by comparing profit estimates across the various RPART model iterations
  - A CP level of 0.0006 yielded the highest profit result (below the logistic model profits)






```
          actual
predicted    0    1
        0 8111  146
        1  707 2088
```

```
rpart_profit_function_test =
rt_prediction_table_test[1,2
rpart_profit_function_test
```

```
[1] 420060
```

Step 5C: Run prediction models – Random Forest

Accuracy:
~92%

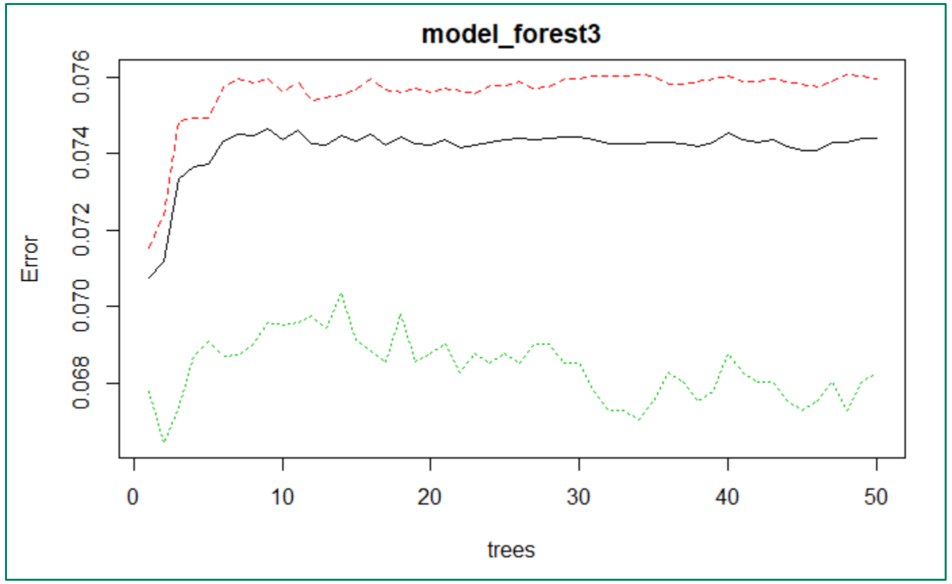
- **Challenge:** insufficient computing power due to the volume of training data we had (>80K observations, number of features and the iterative nature of random forest
- **Solution:** run simplified model by reducing training set (20K observations), removed 'Neighbourhood' variable (as it has many categorical levels) and reduced hyperparameters (e.g. reduced tree / forest complexity)
- **Results:**
  - Similar model accuracy to logistic and RPART models; prediction accuracy driven by **feature engineering and selection** (specifically, 'Prior No Show' variable which proved to have high explanatory power)
  - Slightly **higher profit estimate** than RPART model

```
forest_profit_function_test
```
```

[1] 418806
```

```
forest_prediction_table_test
plot(model_forest)
```
```

         actual
predicted    0    1
        0 8064  190
        1  666 2132
```



Step 6: Choose the most appropriate model

Model chosen: **Random Forest***

Overall prediction accuracy: **92%**

Accuracy of no-show predictions: **76%**

Our model and the profit function we have constructed suggests that, on average, other things being equal, a doctor running a typical 7-hour clinic with 21 appointment slots should **overbook by four appointment slots per clinic**. The specific number will vary by patient characteristics on the day of each clinic and the neighbourhood in which the clinic takes place amongst other factors.

** All three models had similar accuracy, we selected the random forest as it made fewer of the most costly error (i.e., predicting attendance when the patient was actually a no show) and therefore yielded slightly higher profits*

Additional data for further refinement of our models

- There are other factors that are relevant to the likelihood of a no-show but that are not included in the dataset.
  - **Weather conditions** on a particular day that may have contributed to a no-show
  - **The nature of a appointment** (emergency, regular check-up, follow up, new appointment) that could impact the likelihood of a no-show
  - **Traffic conditions** on a particular day that may miss / **vehicle ownership**
  - **Family set-up** / demographics (married, kids, etc.)
  - **Employment** status for an individual patient
  - **Who sets up** the appointment? (Primary user vs. Secondary)