BANK MARKETING

Induced Decision Rules

LUCA FRANCESE 144138
STEFANO FIORINI 144135

WHAT'S BANK MARKETING?

Bank marketing refers to the strategies and techniques used by financial institutions to promote their products and services to customers.

WHY

Attracts New Customers

Builds Trust & Loyalty

Optimizes Resources

COLD CALLING

- Direct Customer Engagement – Allows personalized interaction.
- Immediate Feedback – Sales reps can gauge interest instantly
- Cost-Effective for Lead Generation – Compared to large-scale advertising.
- Upselling & Cross-Selling Opportunities – Existing clients can be targeted for additional products.

WHAT ARE THE COMMON PROBLEMS?

- **Low Conversion Rates** – Many customers **reject** cold calls due to lack of interest.
- **Regulatory & Compliance Issues** – Many countries have **strict telemarketing laws.**
- **Negative Brand Perception** – Unwanted calls can create **annoyance** and harm the bank's reputation.
- **Time-Consuming & Labor-Intensive** – Requires large teams with **low success rates**.
- **Caller Identification Blocking** – Many people **screen or block unknown numbers,** reducing effectiveness.
- **Data Privacy Concerns** – Customers may be reluctant to share personal information over the phone.

WHY USE PREDICTIVE MODELING IN BANK MARKETING?

1. Reduce the number of unnecessary calls by targeting high-probability customers.
2. Lower Cost per Acquisition (CPA) while maintaining or increasing conversions.
3. Use demographics, economic indicators, and past interactions to predict likely subscribers.
4. Avoid excessive calls that can lead to customer fatigue and negative perception.
5. Allocate marketing resources efficiently by prioritizing high-potential leads.

DATA UNDERSTANDING

Observational Unit:
Every single contact with potential customers as part of a marketing campaign.

Desired Target:
Has the client subscribed to a term deposit? (binary)

DATASET OVERVIEW

Source of data:
Portuguese banking institution

Time of collection
January 2012 – December 2012

Number of observations:
45.211

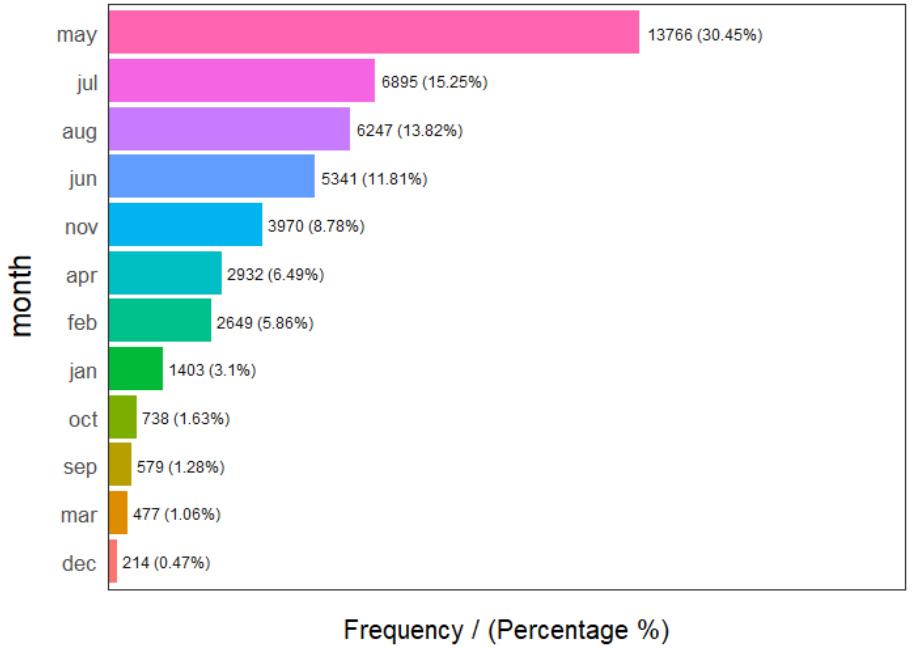
DATA OVERVIEW

Input variables:
  # bank client data:
  1 - age (numeric)
  2 - job : type of job (categorical: "admin.","unknown","unemployed","management","housemaid","entrepreneur","student",
                         "blue-collar","self-employed","retired","technician","services")
  3 - marital : marital status (categorical: "married","divorced","single"; note: "divorced" means divorced or widowed)
  4 - education (categorical: "unknown","secondary","primary","tertiary")
  5 - default: has credit in default? (binary: "yes","no")
  6 - balance: average yearly balance, in euros (numeric)
  7 - housing: has housing loan? (binary: "yes","no")
  8 - loan: has personal loan? (binary: "yes","no")
  # related with the last contact of the current campaign:
  9 - contact: contact communication type (categorical: "unknown","telephone","cellular")
  10 - day: last contact day of the month (numeric)
  11 - month: last contact month of year (categorical: "jan", "feb", "mar", ..., "nov", "dec")
  12 - duration: last contact duration, in seconds (numeric)
  # other attributes:
  13 - campaign: number of contacts performed during this campaign and for this client (numeric, includes last contact)
  14 - pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric, -1 means client was not previously contacted)
  15 - previous: number of contacts performed before this campaign and for this client (numeric)
  16 - poutcome: outcome of the previous marketing campaign (categorical: "unknown","other","failure","success")

  Output variable (desired target):
  17 - y - has the client subscribed a term deposit? (binary: "yes","no")

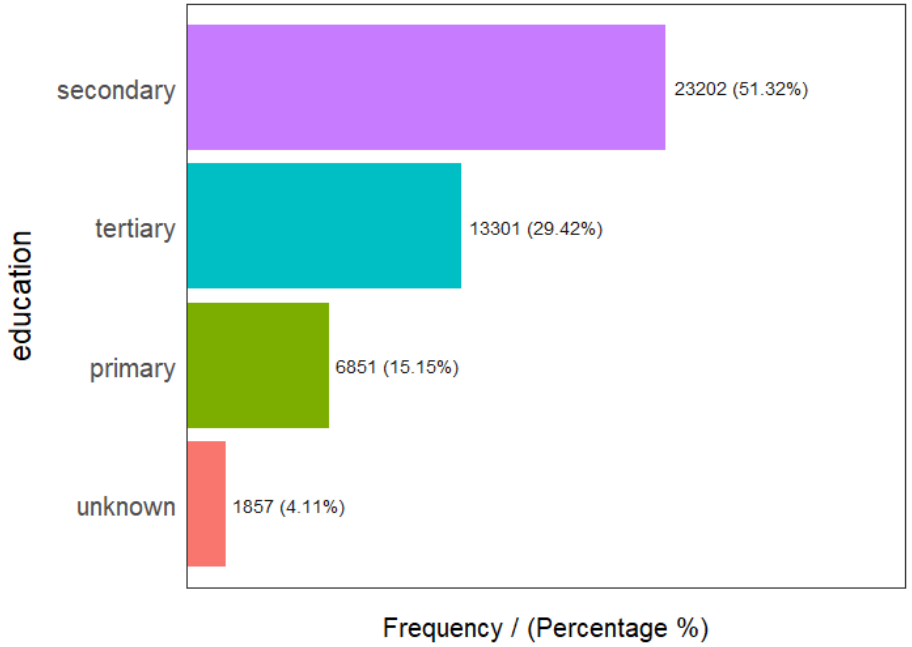
EXPLORATORY DATA ANALYSIS

The chart shows the number and percentage of bank marketing campaigns by month. There is a clear **seasonal pattern**, with most campaigns concentrated in the spring and summer, especially in May, which alone accounts for over 30% of all activity.



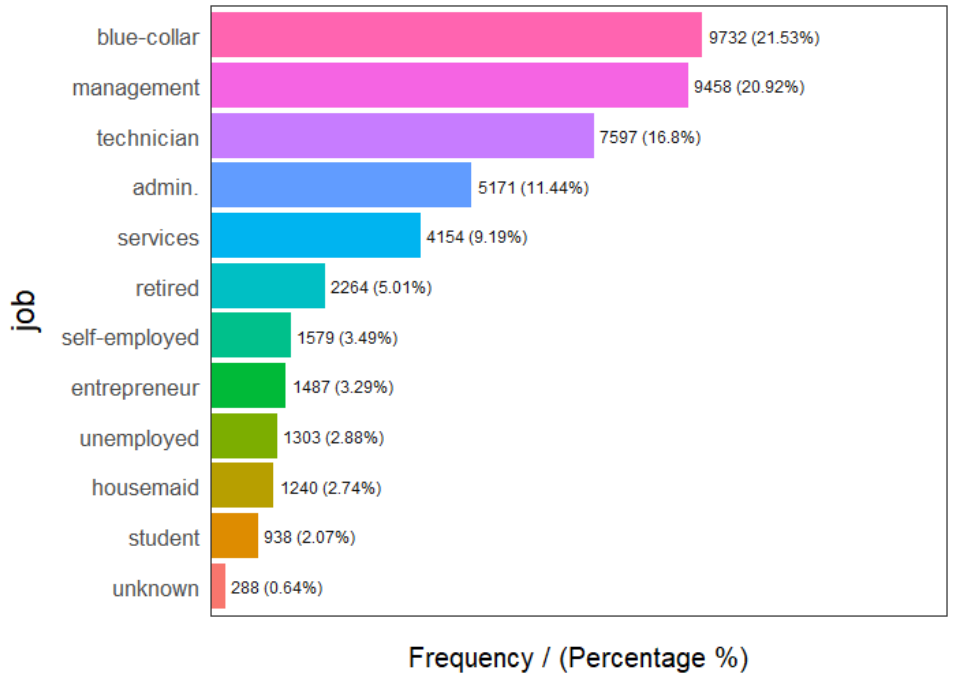
EXPLORATORY DATA ANALYSIS

The chart shows the distribution of education levels among clients in the dataset. **Most clients have a secondary education** (over 51%), followed by those with tertiary (university-level) education (about 29%).



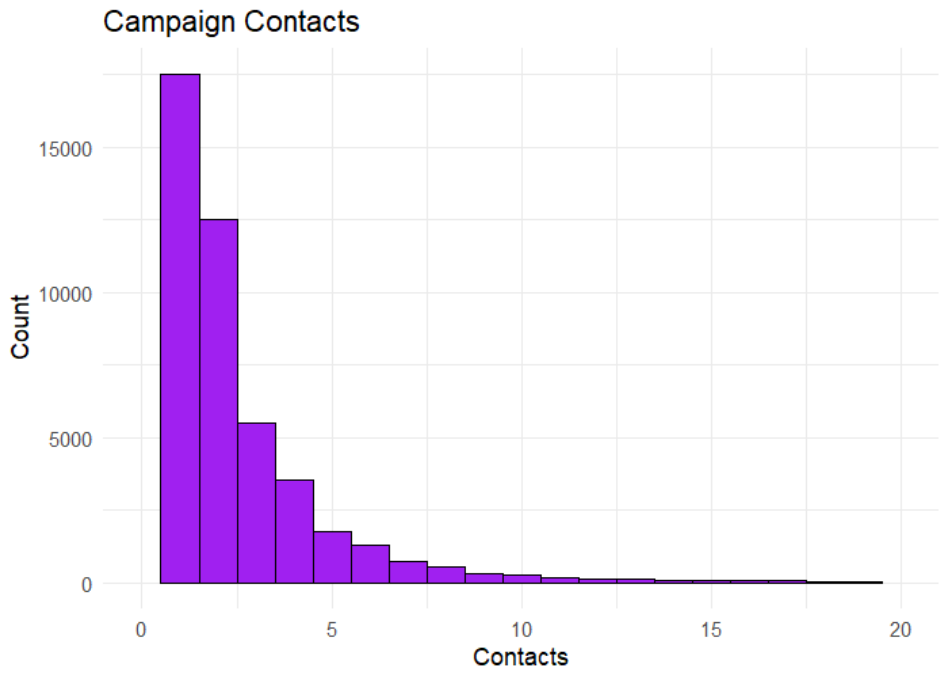
EXPLORATORY DATA ANALYSIS

This chart shows the distribution of clients by job type in the dataset. The largest groups are **blue-collar** (21.5%), **management** (20.9%), and **technician** (16.8%), so these segments will likely have the biggest influence on overall deposit trends.



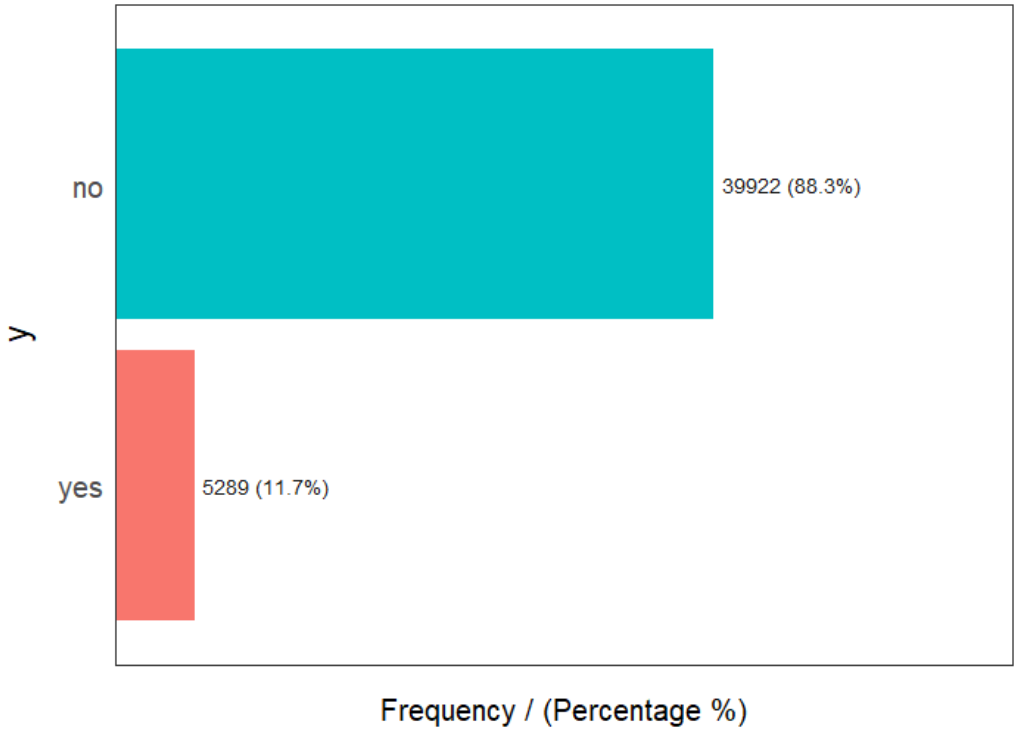
EXPLORATORY DATA ANALYSIS

When we examine the distribution of the "contact" variable, we see that most clients were contacted only once or twice. The frequency drops sharply as the number of contacts increases, with very few clients receiving more than five calls.



EXPLORATORY DATA ANALYSIS

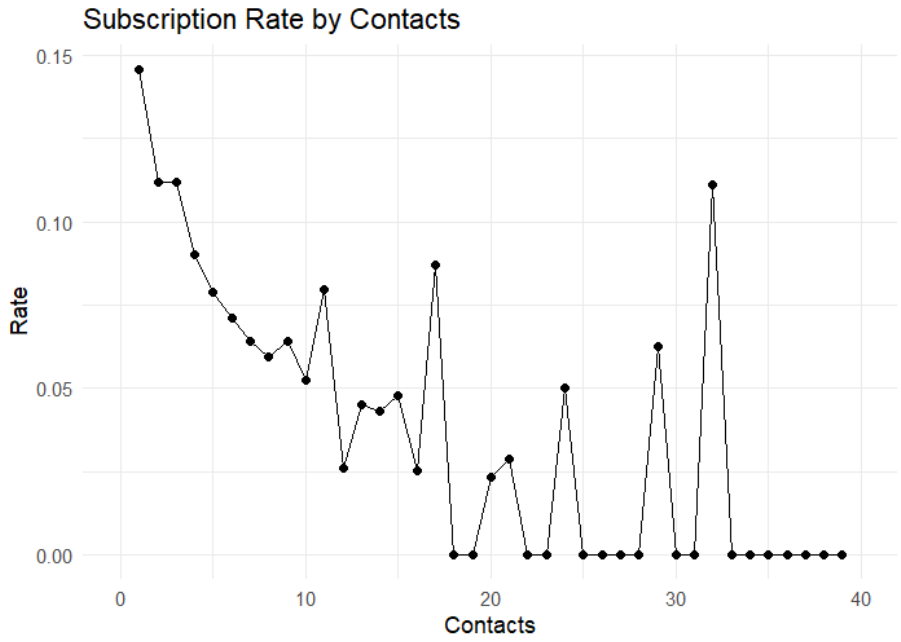
When we analyze the target variable y, which indicates whether clients subscribed to a deposit following the phone calls, we notice a pronounced imbalance in the responses. The vast majority of clients, almost nine out of ten (88.3%), declined the offer, while only a small minority (11.7%) accepted.

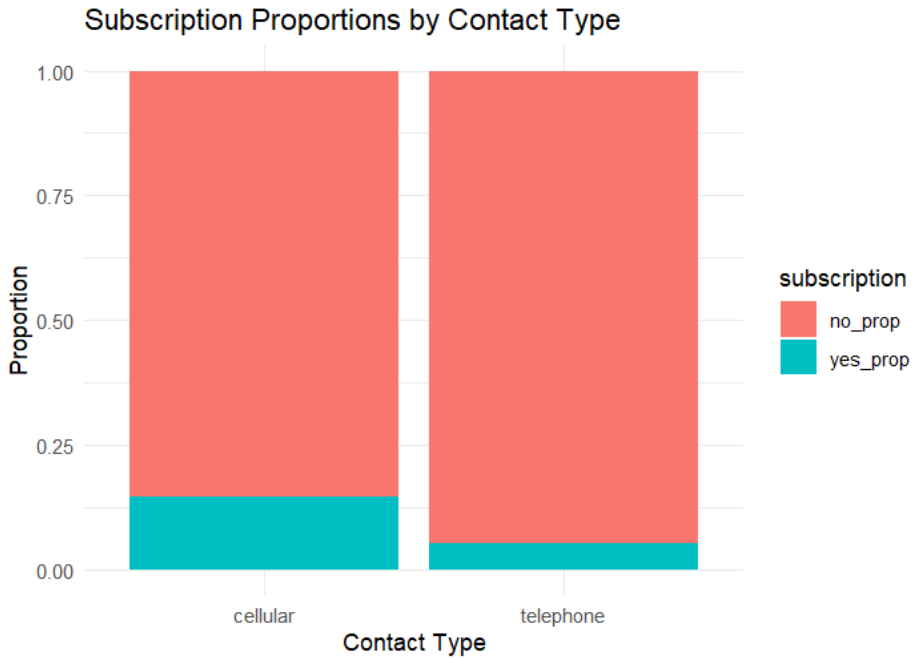


CONTACT RELATIONS

The subscription rate is highest on the first call and drops steadily. After several attempts, the chance of success becomes very low, showing that repeated follow-up calls offer little benefit.

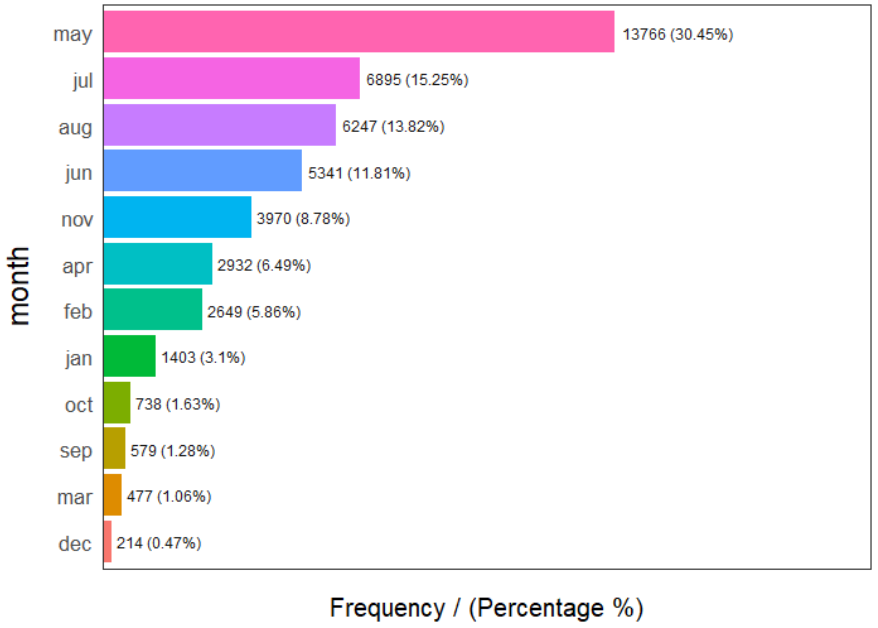
Clients contacted via cellular have a higher subscription rate than those reached by telephone, likely because mobile calls are more direct and effective.

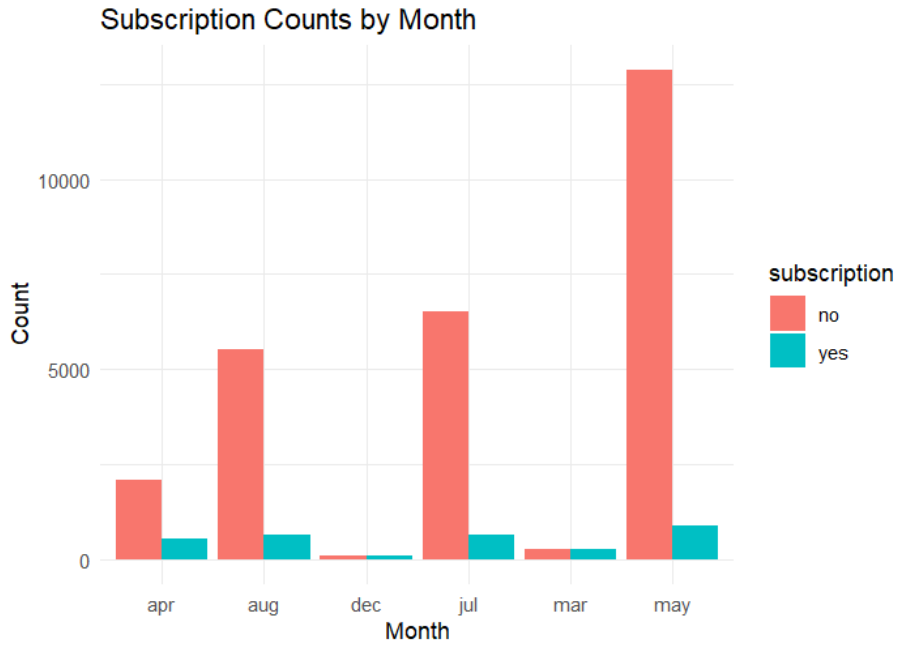




MONTH SUBSCRIPTION COUNTS

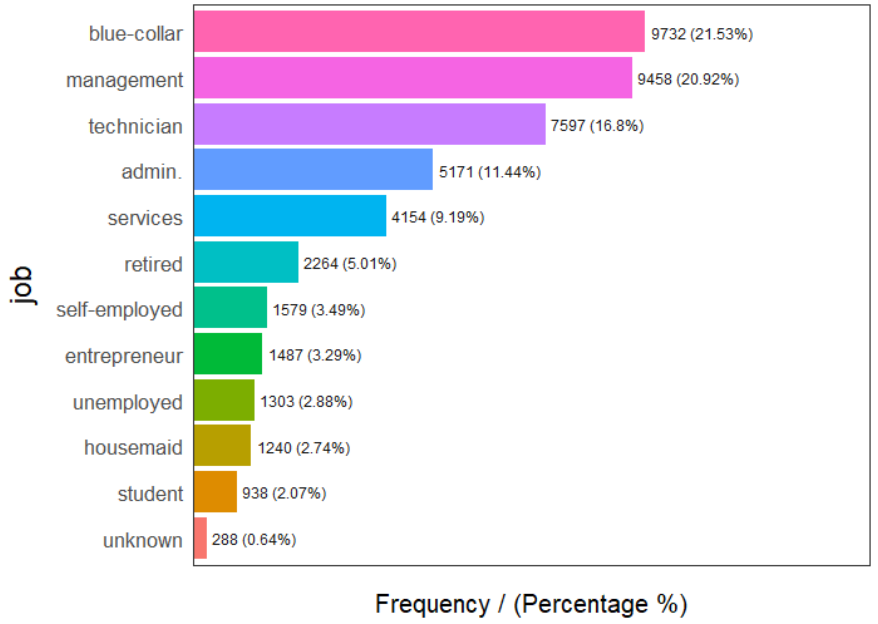
Even though we make far more calls in months like May and July, the number of successful subscriptions increases only slightly. Most clients still say "no", and the conversion rate stays low throughout the year, regardless of how many calls we make.

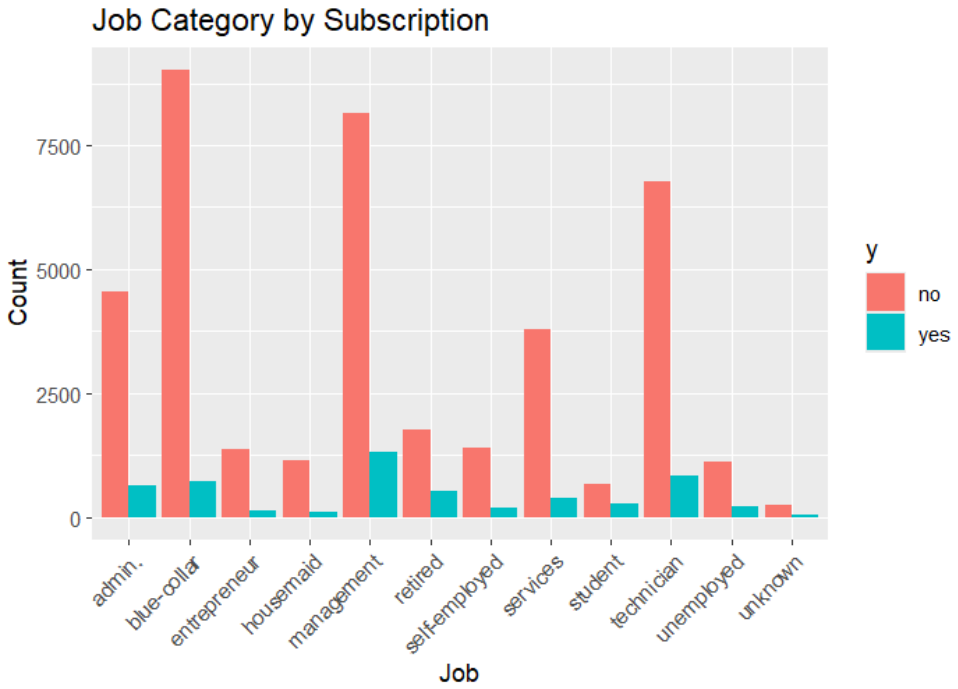




JOB SUBSCRIPTION COUNTS

For all job categories, most clients do not subscribe to a deposit after being contacted. Even in larger groups like management and retired, the number of successful subscriptions remains low compared to the total, showing that job type does not lead to high conversion rates on its own.

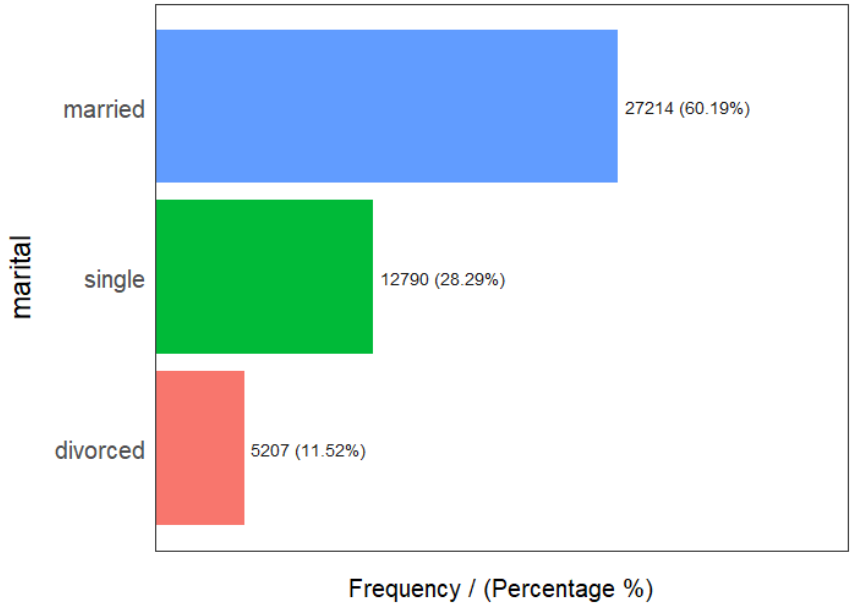


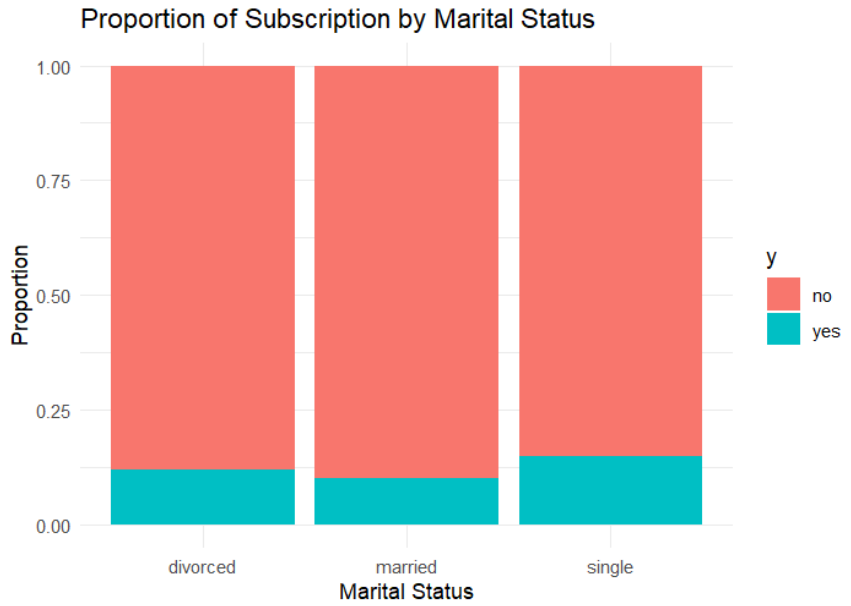


MARITAL RELATIONS

This chart reveals that our client base is predominantly married, with singles making up about 28% and divorced clients just over 11%. This reflects that most of our interactions are with married individuals.

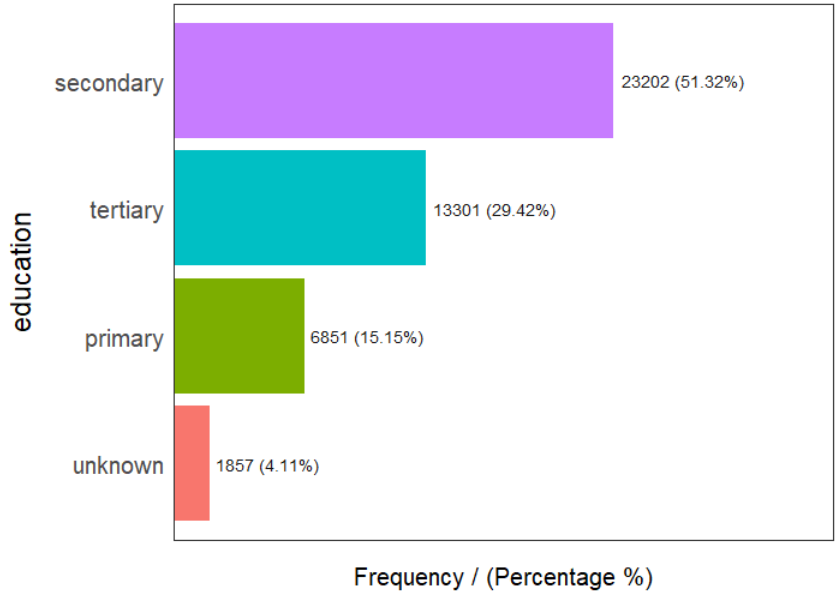
Single clients are more likely to subscribe than married or divorced clients, but overall conversion rates remain low. Focusing on singles or increasing conversions among married clients could improve our results.

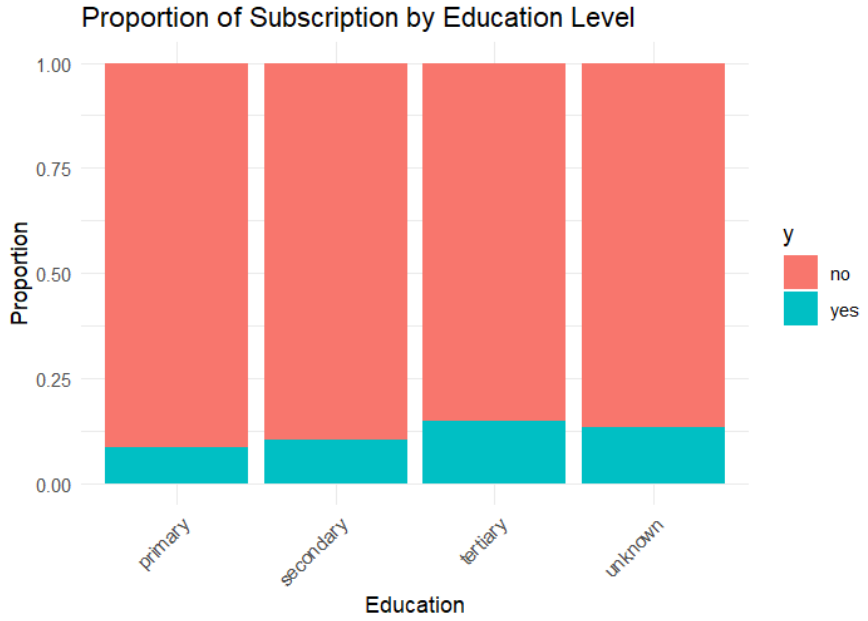




SUBSCRIPTIONS BY EDUCATION LEVEL

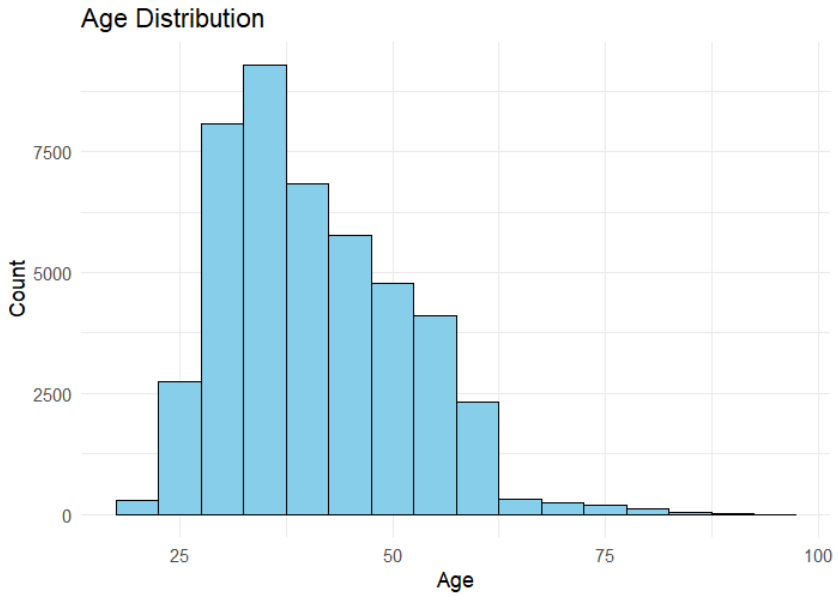
Clients with tertiary education are slightly more likely to subscribe than other groups, but subscription rates remain low across all education levels. Overall, education has only a modest effect on deposit subscriptions.

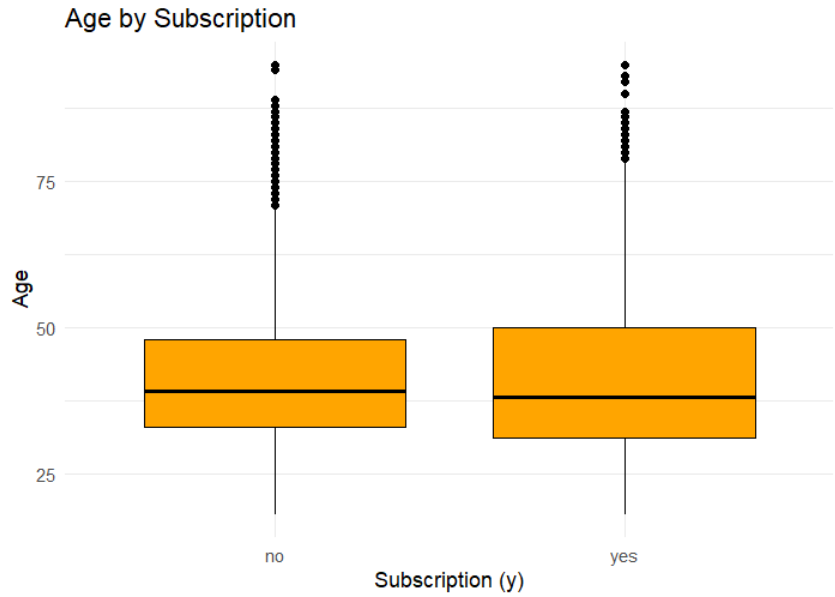




SUBSCRIPTIONS BY AGE

Clients who subscribe to a deposit tend to be slightly older than those who do not, with more diversity and more outliers among older ages in the subscriber group. However, the difference is modest, so age alone is not a strong predictor, its value is greater when combined with other client characteristics.

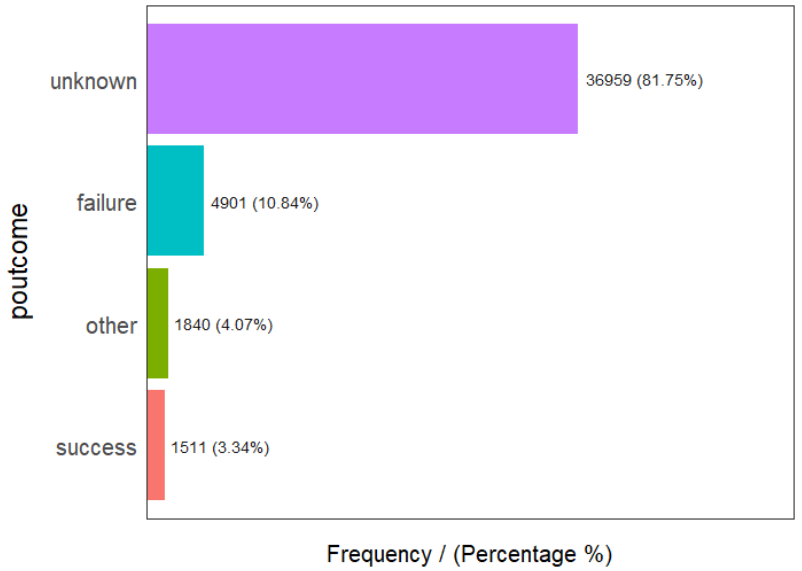




DATA PREPARATION

We exclude the duration variable because it is only known after the call ends and reveals the outcome, causing data leakage. Using it would give unrealistic predictions, since this information isn't available before making the call.

We are excluding both 'poutcome' and 'pdays' because over 80% of 'poutcome' values are missing, making the variable unreliable. Since 'pdays' is directly linked to 'poutcome', it also lacks meaningful information, so removing both ensures our analysis remains robust and accurate.



DATA PREPARATION

We replaced all the “unknown” values with NA and removed those records, ensuring our analysis is based only on complete, reliable data and minimizing bias from missing or unclear information.

```
bank_clean <- bank %>%
mutate(across(everything(),
~ifelse(tolower(.x) == "unknown", NA, .x)))
colSums(is.na(bank_clean))
bank_clean <- na.omit(bank_clean)
```

UNDERSAMPLING

Train undersampling

```
min_n <- min(table(train$y))

train_balanced <- train %>%
group_by(y) %>%
sample_n(min_n) %>%
ungroup()
```

Test undersampling

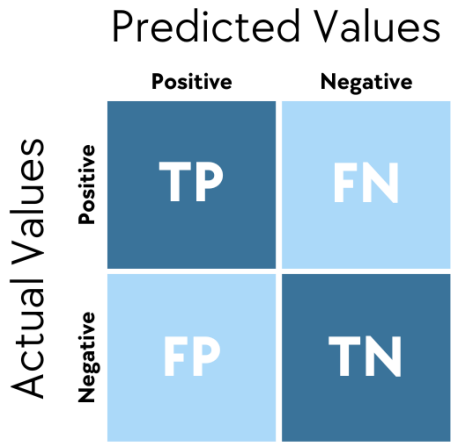
```
min_n_test <- min(table(test$y))

test_balanced <- train %>%
group_by(y) %>%
sample_n(min_n_test) %>%
ungroup()
```

OBJECTIVE OF THE MODELLING PHASE

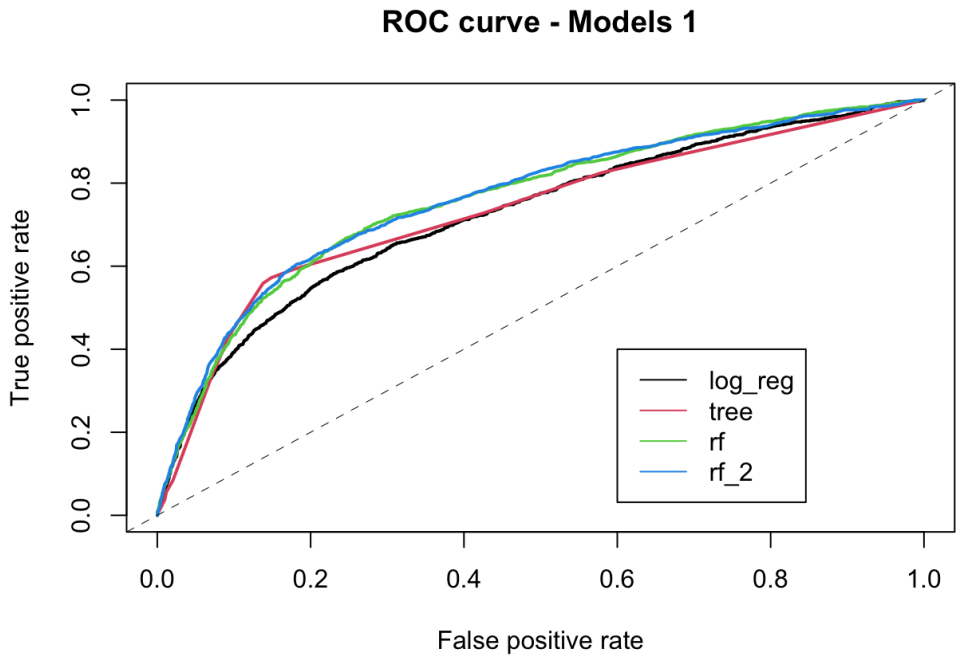
Relevant metrics

Metrics should be aligned with the goal of not losing potentially interested customers.

- Low False Omitted Rate (FN/ total predicted negative)
- High F1-score (2TP/(2TP + FP + FN)
- High Sensitivity (TP / actual positive)



Classification models 1

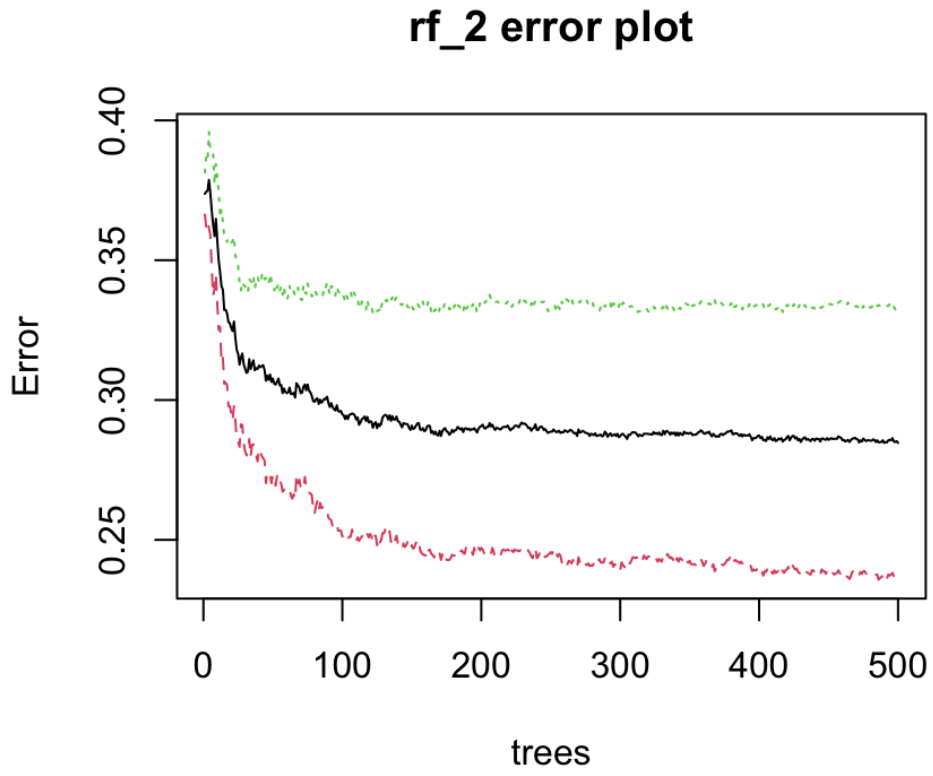


AUC Values:

log_reg: 0.7246397
tree: 0.7321444
rf: 0.7601387
rf_2: 0.7628565

	log_reg	tree	rf	rf_2
accuracy	0.7283052	0.8121072	0.7499173	0.7456169
MER	0.2716948	0.1878928	0.2500827	0.2543831
precision	0.2808118	0.3866525	0.3136739	0.3091448
sensitivity	0.596395	0.5721003	0.6543887	0.6543887
specificity	0.7499038	0.8514051	0.7655588	0.7605543
FOR	0.3064947	0.1612589	0.2851569	0.2930276
F1	0.3818364	0.4614412	0.4240731	0.4199145

Random forest – cross validation

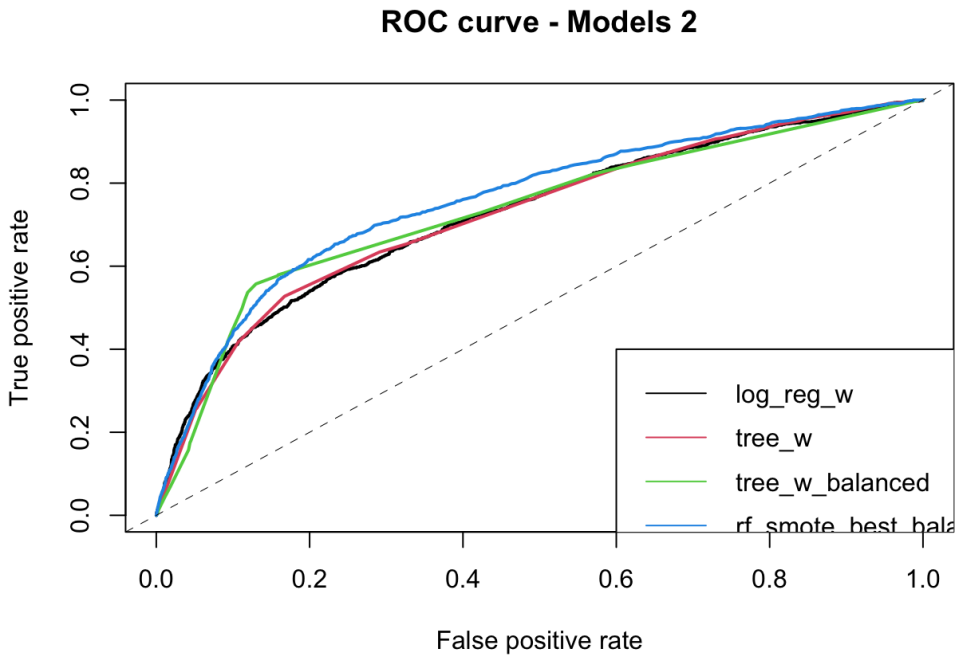



```
train_control <- trainControl(method = "cv", number = 5)

rf_2 <- train(y ~ ., data = train_balanced, method = "rf", trControl = train_control)
```

The training data is split into 5 equal parts:

- The model is trained on 4 parts and tested on the 5th.
- This is repeated till each fold is used once for validation.
- The performance metrics are averaged to evaluate each parameter combination.

Classification models 2



AUC Values:

log_reg_w: 0.7238095
tree_w: 0.7239257
tree_w_balanced: 0.7318169
rf_smote_best_balanced: 0.7581328

	log_reg_w	tree_w	tree_w_balanced	rf_smote_best_balanced
accuracy	0.8608446	0.2942993	0.8323961	0.7506892
MER	0.1391554	0.7057007	0.1676039	0.2493108
precision	0.5216049	0.1595801	0.4244114	0.3143611
sensitivity	0.1324451	0.9412226	0.5368339	0.653605
specificity	0.9801104	0.1883742	0.8807905	0.7665854
FOR	0.01772441	4.099157	0.1246144	0.28351
F1	0.21125	0.2728925	0.4740484	0.4245355

Decision trees – Models 2

Trained with all data

```
tree_w <- rpart(y ~ ., data = train,
method = "class",
cp = 0.0024,
parms = list(split =
"information", prior = c(0.25, 0.75)))
```

Trained with balanced data

```
tree_w_balanced <- rpart(y ~ ., data =
train_balanced, method = "class",
cp = 0.0024,
parms = list(split =
"information",prior = c(0.7, 0.3)))
```

Random forest – Models 2

Oversampling - SMOTE

```
rec <- recipe(y ~ ., data = train) %>%
step_dummy(all_nominal_predictors()) %>%
step_smote(y) %>%
prep()

# Apply the recipe to get processed data
train_processed <- juice(rec)
```

We are preparing rec in order to be used in the tuning process with “caret” package

Decision trees – Models 2

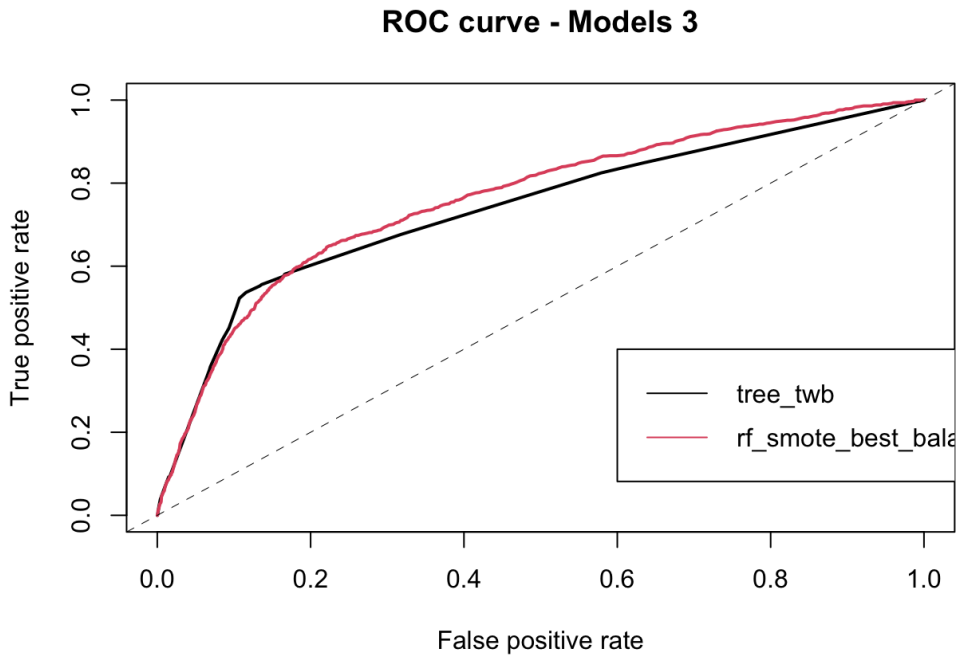
Tuning step

```
rf_smote_balanced <- caret::train(
x = X_train_bal,
y = y_train_bal,
method = "ranger",
trControl = train_control,
tuneGrid = expand.grid(
mtry = c(2, 4, 6, 8),
splitrule = "gini",
min.node.size = c(1, 5, 10)
)
)
```

Tuned model

```
rf_smote$bestTune

rf_smote_best_balanced <-
randomForest(
y ~ .,
data = train_balanced,
mtry = 6,
nodesize = 10,
importance = TRUE
)
```

Classificstion models 3



AUC Values:

tree_twb: 0.7376893

rf_smote_best_balanced_less: 0.7598399

	tree_twb	rf_smote_best_balanced_less
accuracy	0.8407763	0.7500276
MER	0.1592237	0.2499724
precision	0.4440746	0.3140713
sensitivity	0.5227273	0.6559561
specificity	0.8928526	0.7654305
FOR	0.1103476	0.2854466
F1	0.4802016	0.4247653

Importance of variables – Models 3

Classification tree

threshold <- 0.10 *
max(importance_twb)

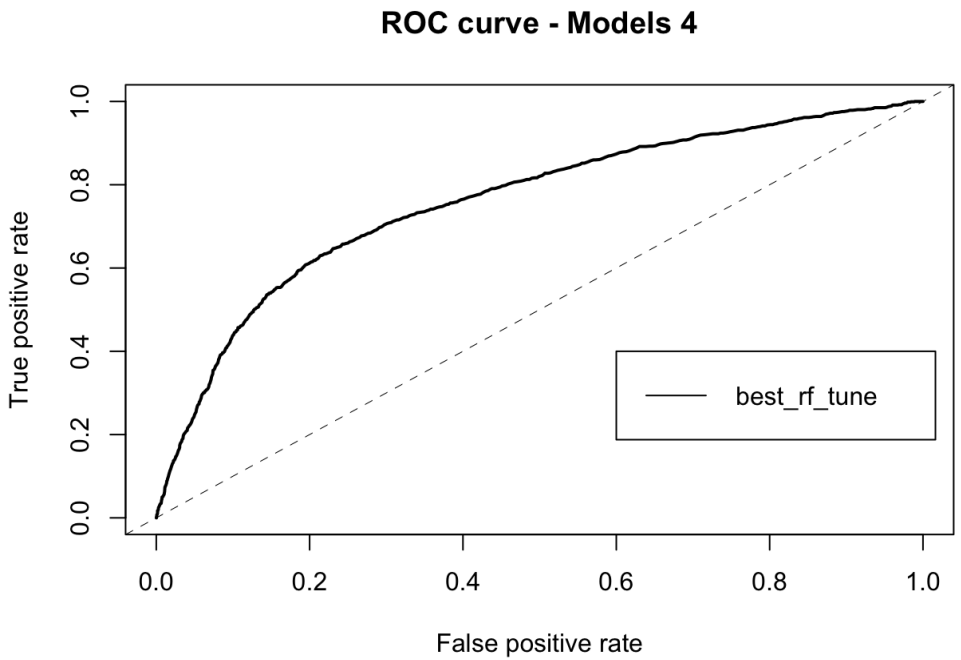
selected_vars <-
names(importance_twb[importance_t
wb >= threshold])

Random forest

varImp(rf_smote_best_balanced)

Throughout a trials and errors process
we decided for
y ~ .- default - job

Classification models 4



AUC Values:

best_rf_tune : 0.7588007

	best_rf_tune
accuracy	0.7429706
MER	0.2570294
precision	0.3069887
sensitivity	0.6575235
specificity	0.7569614
FOR	0.2989268
F1	0.4185582

Random forest – Models 4

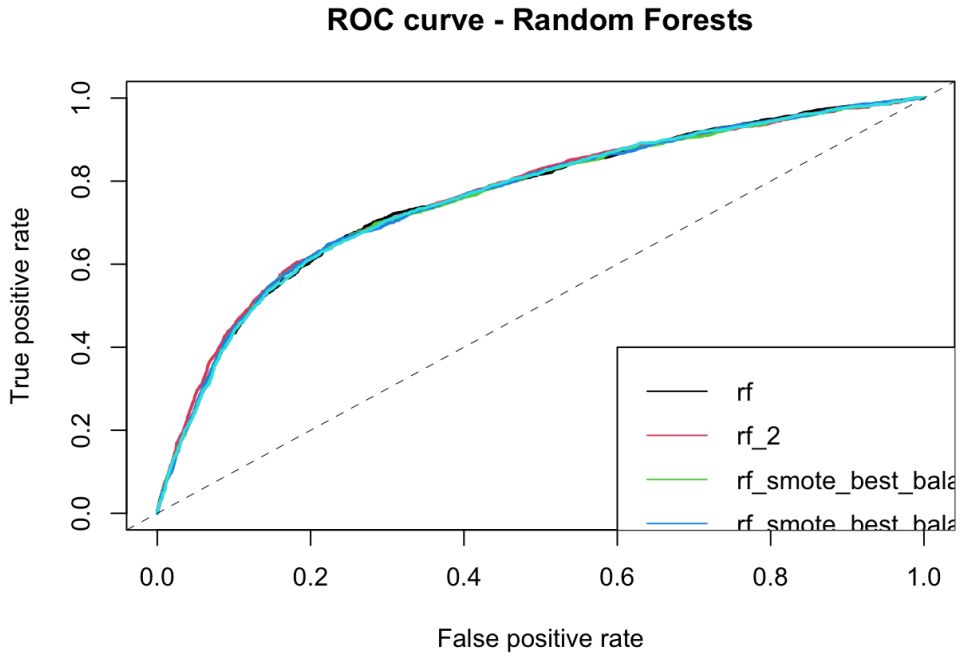
Tuning with “mlr”

```
rf_tune <- tuneParams(learner =
                      rf_to_be,
                    resampling = set_rf_cv,
                      task = train_task,
                    par.set = rf_param,
                      control = rf_cntl,
                      measure = acc)
```

makeLearner function

```
rf_to_be <-
makeLearner("classif.randomForest",
                    predict.type = "response",
                     par.vals = list(ntree =
floor(0.1*nrow(train_balanced_mutate)
),
                                     mtry =
floor((ncol(train_balanced_mutate)-
1)/3)))
rf$par.vals <- list(importance = TRUE)
```

Random forests



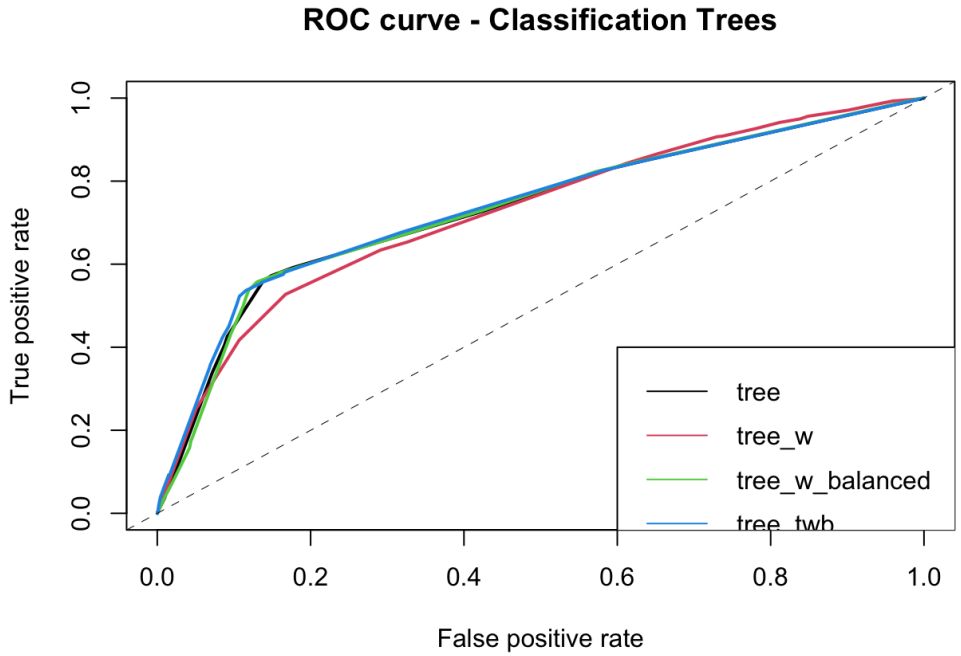
AUC Values:

rf : 0.7601387
rf_2 : 0.7628565
rf_smote_best_balanced : 0.7581328
rf_smote_best_balanced_less : 0.7598399
best_rf_tune : 0.7588007

The model that best represents our needs is "rf_smote_best_balanced_less".

	rf	rf_2	rf_smote_best_balanced	rf_smote_best_balanced_less	best_rf_tune
accuracy	0.7496968	0.7453964	0.7509097	0.7501378	0.7429706
MER	0.2503032	0.2546036	0.2490903	0.2498622	0.2570294
precision	0.3132983	0.3087745	0.3144583	0.3143286	0.3069887
sensitivity	0.653605	0.653605	0.6528213	0.6567398	0.6575235
specificity	0.7654305	0.760426	0.7669704	0.7654305	0.7569614
FOR	0.2853129	0.2931847	0.282866	0.2854912	0.2989268
F1	0.4235653	0.4194116	0.4244586	0.4251649	0.4185582

Classification trees



AUC Values:

tree: 0.7321444
tree_w: 0.7239257
tree_w_balanced: 0.7318169
tree_twb: 0.7376893

The model that best represents our needs is "tree_twb".

	tree	tree_w	tree_w_balanced	tree_twb
accuracy	0.8121072	0.2942993	0.8323961	0.8407763
MER	0.1878928	0.7057007	0.1676039	0.1592237
precision	0.3866525	0.1595801	0.4244114	0.4440746
sensitivity	0.5721003	0.9412226	0.5368339	0.5227273
specificity	0.8514051	0.1883742	0.8807905	0.8928526
FOR	0.1612589	4.099157	0.1246144	0.1103476
F1	0.4614412	0.2728925	0.4740484	0.4802016

Model decision

```
> CM_6[["rf_smote_best_balanced_less"]]

          no   yes
  no    5965   438
  yes   1828   838
> CM_5[["tree_twb"]]

          no   yes
  no    6958   609
  yes    835   667
```

Final choice:
Considering our mostly considered metrics (sensitivity, FNR, FOR) , the final chosen model will be "tree_twb".

Business implications

Mean duration of the previous call when the outcome of that previous call is "failure": 244.2 seconds.

Average hourly gross salary of a bank call center operator in Portugal: 11.15€.

Predicted negative observations: 7,793.

Total hours that could be saved: 528.6 hours.

Economic value saved: 5893,89€.

Source: SalaryExpert

THANK YOU FOR YOUR ATTENTION

LUCA FRANCESE 144138
STEFANO FIORINI 144135