

**R (ADP)**

2023-05-18

# Table of contents

Preface	3
1 Introduction	4
1 ADP :	5
2 27 ADP	7
1.	8
2.	12
3.	15
4.	19
	22
3 Summary	23
References	24

# Preface

This is a Quarto book.

To learn more about Quarto books visit <https://quarto.org/docs/books>.

1 + 1

[1] 2

# 1 Introduction

This is a book created from markdown and executable code.

See Knuth (1984) for additional discussion of literate programming.

```
kmeans_spec <- linear_reg()
show_engines("linear_reg")
```

```
# A tibble: 7 x 2
  engine mode
  <chr>   <chr>
1 lm      regression
2 glm     regression
3 glmnet  regression
4 stan    regression
5 spark   regression
6 keras   regression
7 brulee  regression
```

## 1.0.0.1

```
yardstick::metric_set()
```

## **Part I**

**ADP :**

R ADP

**2 27 ADP**

# 1.

## 1.1. (EDA) .

```
pacman::p_load(tidyverse, tidymodels, data.table, gt,  
               skimr, ggcorrplot, themis, solitude)
```

```
db_1a <- fread("test/27/data/problem1.csv")
```

(EDA) skim .

```
db_1a %>% skim()
```

Table 2.1: Data summary

Name	Piped data
Number of rows	1193
Number of columns	20
Key	NULL
Column type frequency:	
numeric	20
Group variables	None

### Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Time	0	1	91514.4947896086	0.00	50265.0081797001	36995.00172676	0.00			



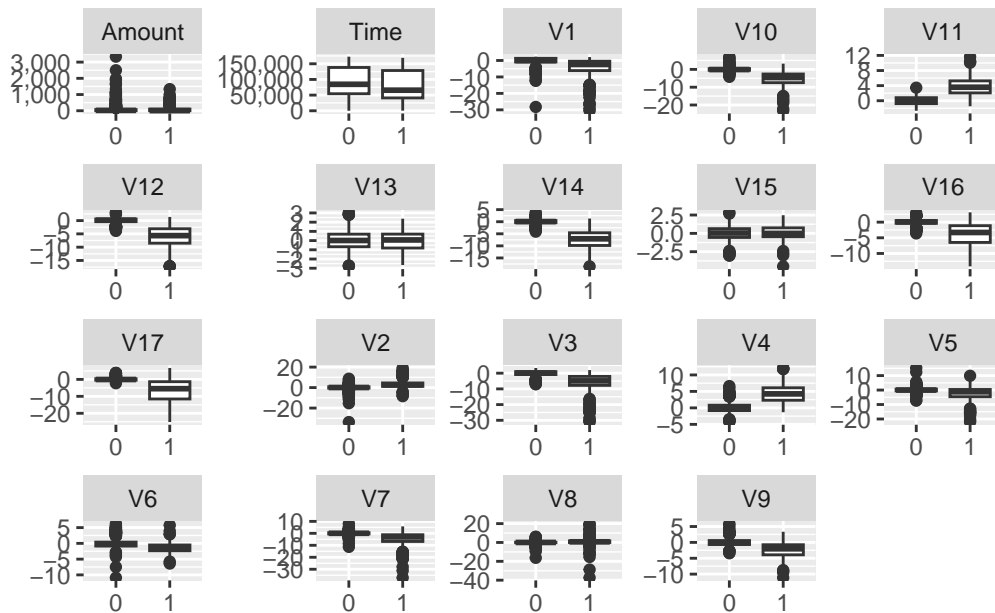
skim_variable	n_missing	n_complete	mean	sd	p0	p25	p50	p75	p100	hist
V1	0	1	-0.76	3.62	-	-1.30	-0.30	1.25	2.32	
V2	0	1	0.54	2.66	-	-0.44	0.23	1.11	19.17	
V3	0	1	-1.15	3.91	-	-1.61	-0.23	0.81	3.32	
V4	0	1	0.78	2.35	-	-0.61	0.33	1.34	12.11	
V5	0	1	-0.41	2.70	-	-0.84	-0.07	0.68	15.28	
V6	0	1	-0.28	1.51	-	-1.02	-0.38	0.30	6.27	
V7	0	1	-0.85	3.40	-	-0.81	-0.08	0.46	8.12	
V8	0	1	0.15	2.47	-	-0.22	0.07	0.45	20.01	
V9	0	1	-0.45	1.66	-	-1.04	-0.21	0.45	5.92	
V10	0	1	-0.90	2.90	-	-0.88	-0.23	0.29	7.14	
V11	0	1	0.66	1.96	-	-0.60	0.28	1.21	11.67	
V12	0	1	-1.01	3.00	-	-0.97	-0.01	0.51	3.11	
V13	0	1	0.01	1.01	-	-0.69	0.02	0.70	3.07	
V14	0	1	-1.17	3.23	-	-0.95	-0.10	0.42	3.89	
V15	0	1	0.02	0.91	-	-0.55	0.05	0.66	2.87	
V16	0	1	-0.63	2.27	-	-0.70	-0.06	0.46	3.14	
V17	0	1	-1.07	3.81	-	-0.68	-0.15	0.37	6.74	
Amount	0	1	88.89	220.14	0.00	3.54	20.99	77.49	3335.73	
Class	0	1	0.17	0.37	0.00	0.00	0.00	0.00	1.00	

:

- 
- (numeric) , Class (factor)

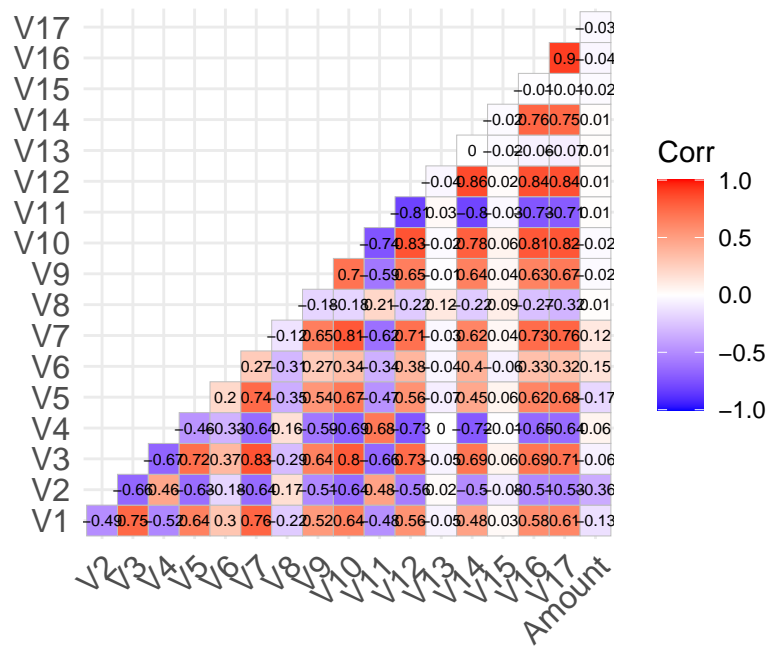
- Class 1 16.7%

```
db_1a %>%
  # id
  mutate(id = row_number()) %>%
  # Wide-to-long
  pivot_longer(cols = !c(id, Class)) %>%
  ggplot(aes(x = as.factor(Class), y = value)) +
  geom_boxplot() +
  facet_wrap(name ~ ., scales = "free") +
  labs(x = "", y = "") +
  scale_y_continuous(labels = comma)
```



## 1.2.

```
db_1a %>% select(-c("Time", "Class")) %>%
  cor() %>% ggcorrplot(type = "lower", lab = TRUE, lab_size = 2)
```



V1-V3, V14-V17

0.75

.

.

## 2.

### 2.1. 2 .

PCA , MDS , t-SNE . , t-SEN

PCA .

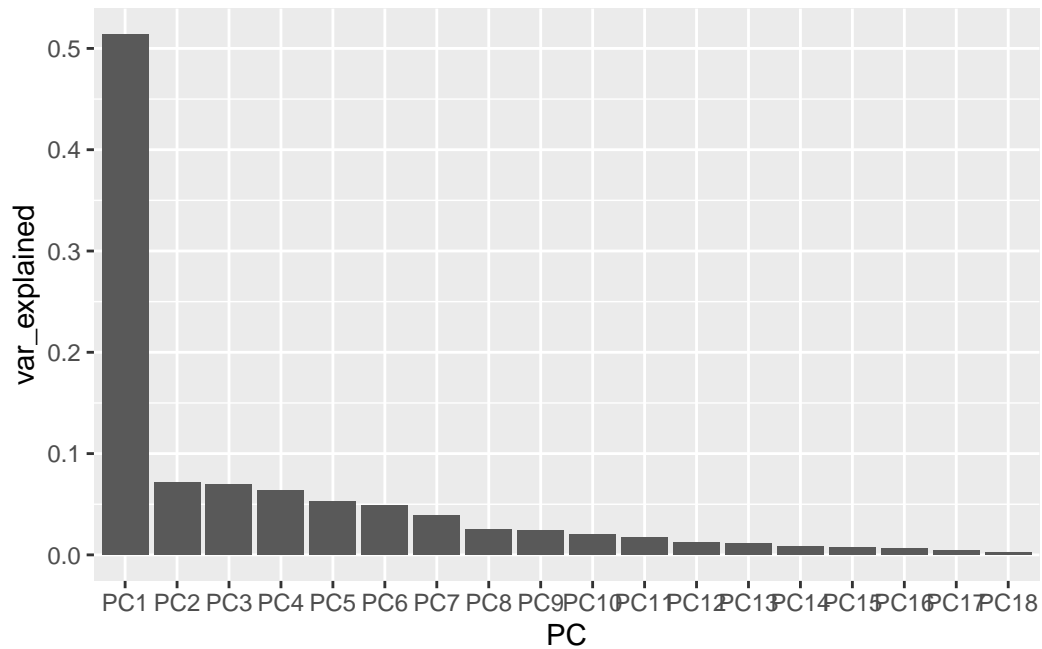
PCA scree .

```
db_1b <- db_1a %>% select(-c("Time", "Class"))

# PCA recipe
rec <- recipe(~., data = db_1b) %>%
  step_center(all_predictors()) %>%
  step_scale(all_predictors()) %>%
  step_pca(all_predictors())

# recipe ( )
prep_rec <- prep(rec)

# Scree
sdev <- prep_rec$steps[[3]]$res$sdev
percent_variation <- sdev^2 / sum(sdev^2)
data.frame(PC=paste0("PC",1:length(sdev)),
            var_explained=percent_variation,
            stringsAsFactors = FALSE) %>%
  mutate(PC = fct_inorder(PC)) %>%
  ggplot(aes(x=PC,y=var_explained))+geom_col()
```



2        1                                2        .

2    PCA                                .

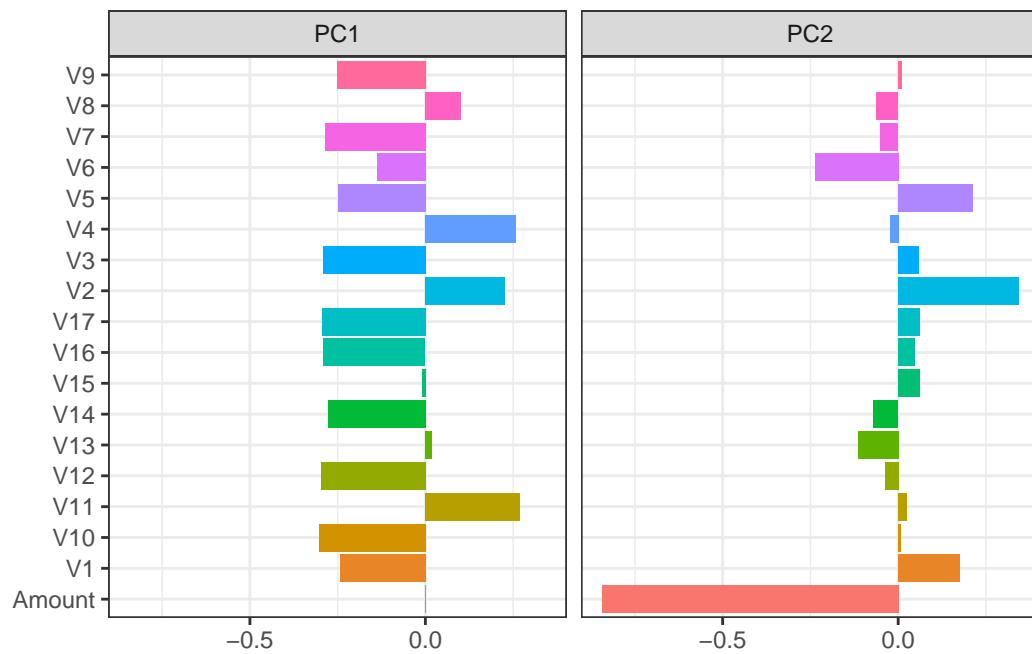
```
db_1b <- db_1a %>% select(-c("Time", "Class"))
# PCA    recipe
rec <- recipe(~., data = db_1b) %>%
  step_center(all_predictors()) %>%
  step_scale(all_predictors()) %>%
  step_pca(all_predictors(), num_comp = 2) #    2

prep_rec <- prep(rec)

tidy_rec <- tidy(prepare_rec, number = 3)

tidy_rec %>%
  filter(component %in% paste0("PC", 1:2)) %>%
  mutate(component = fct_inorder(component)) %>%
  ggplot(aes(value, terms, fill = terms)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~component, nrow = 1) +
  labs(y = NULL, x = NULL) +
```

theme\_bw()



## 3.

### 3.1.

- : ,
- :

- : ,
- :

. Class 20 , .

```
table(db_1a$Class)
```

```
0 1  
993 200
```

### 3.2. 2 .

, 1) SMOTE 2) upsampling .

```
db_1c <- db_1a[, -c("Time")]
```

```
# Recipe
```

```
rec <- recipe(Class ~ ., data = db_1c) %>%  
  step_mutate(Class = as.factor(Class)) %>%  
  step_center(all_predictors(), -all_outcomes()) %>%  
  step_scale(all_predictors(), -all_outcomes()) %>%  
  step_pca(all_predictors(), -all_outcomes(), num_comp = 2) # 2
```

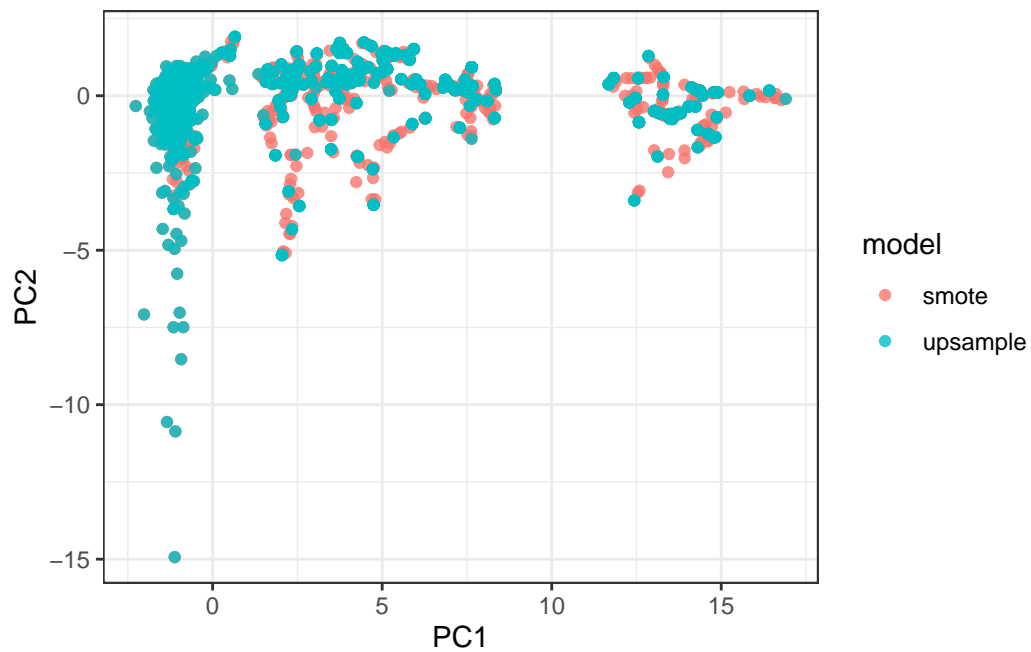
```

#
model_spec <- decision_tree() %>%
  set_engine("rpart") %>%
  set_mode("classification")

# SMOTE
rec_smote <- rec %>% step_smote(Class) %>% prep(db_1c) %>% bake(new_data = NULL)
rec_upsample <- rec %>% step_upsample(Class) %>% prep(db_1c) %>% bake(new_data = NULL)

bind_rows(rec_smote %>% mutate(model = "smote"),
          rec_upsample %>% mutate(model = "upsample")) %>%
  ggplot(aes(x = PC1, y = PC2, color = model)) +
  geom_point(alpha = 0.8) +
  theme_bw()

```



,          upsample          SMOTE          .



### 3.3.

```
set.seed(2023)

db_1c <- db_1a[, -c("Time")]

# Recipe
rec <- recipe(Class ~ ., data = db_1c) %>%
  step_mutate(Class = as.factor(Class)) %>%
  step_center(all_predictors(), -all_outcomes()) %>%
  step_scale(all_predictors(), -all_outcomes()) %>%
  step_pca(all_predictors(), -all_outcomes(), num_comp = 2) # 2

#
model_spec <- decision_tree() %>%
  set_engine("rpart") %>%
  set_mode("classification")

# SMOTE
rec_smote <- rec %>% step_smote(Class) %>% prep(db_1c)
wk_smote <- workflow() %>% add_recipe(rec_smote) %>% add_model(model_spec)

cv <- vfold_cv(db_1c)

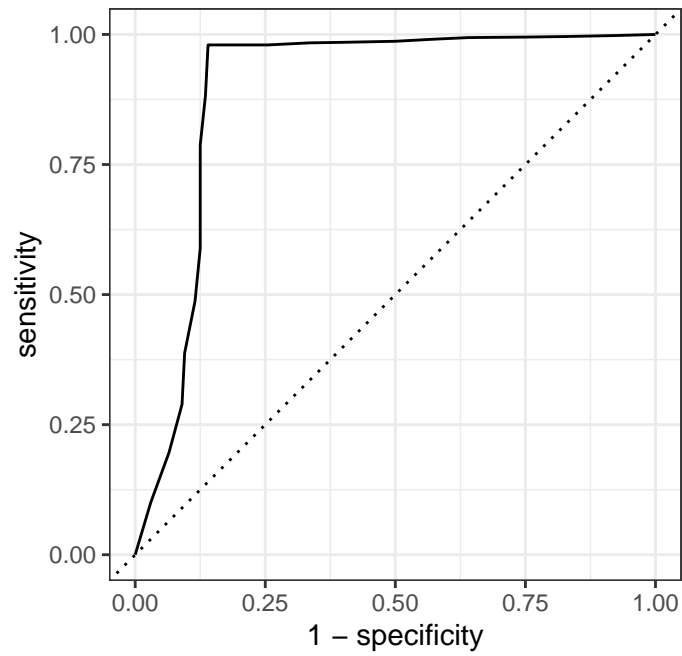
result_smote <- fit_resamples(
  wk_smote,
  resamples = cv,
  control = control_resamples(save_pred = TRUE))

predictions <- result_smote %>%
  collect_predictions()

collect_metrics(result_smote)

# A tibble: 2 x 6
  .metric .estimator mean      n std_err .config
  <chr>   <chr>    <dbl> <int>   <dbl> <chr>
1 accuracy binary    0.960   10 0.00671 Preprocessor1_Model1
2 roc_auc  binary    0.922   10 0.0185  Preprocessor1_Model1
```

```
predictions %>%
  yardstick::roc_curve(Class, .pred_0) %>% autoplot()
```



, 95.9%, AUC 92.1% .

## 4.

### 2.1 4.1. 2 .

- Isolation Forest: , . ,  
– : ,  
– :  
• Autoencoder: , . , Autoencoder , ,  
– :  
– :

### 2.2 4.2. 3

```
db_1c <- db_1a[, -c("Time")]

# Recipe
rec <- recipe(Class ~ ., data = db_1c) %>%
  step_mutate(Class = as.factor(Class)) %>%
  step_center(all_predictors(), -all_outcomes()) %>%
  step_scale(all_predictors(), -all_outcomes()) %>%
  step_pca(all_predictors(), -all_outcomes(), num_comp = 2) # 2

# SMOTE
rec_smote <- rec %>% step_smote(Class) %>% prep(db_1c)

wk_smote <- workflow() %>% add_recipe(rec_smote) %>% add_model(model_spec)

data_preprocessed <- juice(rec_smote)
```

```

model <- isolationForest$new()
model$fit(data_preprocessed[, -which(names(data_preprocessed) == "Class")])

```

```

INFO [17:21:44.251] dataset has duplicated rows
INFO [17:21:44.270] Building Isolation Forest ...
INFO [17:21:44.297] done
INFO [17:21:44.298] Computing depth of terminal nodes ...
INFO [17:21:44.491] done
INFO [17:21:44.536] Completed growing isolation forest

```

```

scores <- model$predict(data_preprocessed[, -which(names(data_preprocessed) == "Class")])
data_preprocessed$anomaly_score <- scores

quantile(data_preprocessed$anomaly_score$anomaly_score)

```

```

      0%      25%      50%      75%     100%
0.5820092 0.5827973 0.5847722 0.5955529 0.7654805

```

```

#
threshold <- 0.6
data_preprocessed$anomaly <- ifelse(data_preprocessed$anomaly_score > threshold, 1, 0)

# SMOTE
ttt <- as.data.table(data_preprocessed$anomaly)

data_preprocessed$anomaly <- as.factor(ttt$anomaly_score)
table(data_preprocessed$Class, data_preprocessed$anomaly)

```

```

      0      1
0 929   64
1 649  344

```

```

tttt <- data_preprocessed %>%
  conf_mat(truth = Class, estimate = anomaly)

class_metrics <- metric_set(accuracy, sens, spec)

```

```
data_preprocessed |>
  class_metrics(truth = Class, estimate = anomaly)
```

```
# A tibble: 3 x 3
  .metric .estimator .estimate
  <chr>    <chr>        <dbl>
1 accuracy binary      0.641
2 sens     binary      0.936
3 spec     binary      0.346
```

- <https://amaruak00.github.io/2022/12/17/adp27th.html>
- <https://www.datamanim.com/dataset/ADPpb/00/27.html>
- <https://www.kaggle.com/code/johyunkang/adp-27/notebook>

## 3 Summary

In summary, this book has no content whatsoever.

**1** + **1**

[1] 2

## References

Knuth, Donald E. 1984. “Literate Programming.” *Comput. J.* 27 (2): 97–111. <https://doi.org/10.1093/comjnl/27.2.97>.