Challenge 5- LIME

2024-21-06

Contents

```
library(h2o)
library(recipes)
library(readxl)
library(tidyverse)
library(tidyquant)
library(lime)
library(rsample)
```

```
process_hr_data_readable <- function(data, definitions_tbl) {

    definitions_list <- definitions_tbl %>%
        fill(...1, .direction = "down") %>%
        filter(!is.na(...2)) %>%
        separate(...2, into = c("key", "value"), sep = " '", remove = TRUE) %>%
        rename(column_name = ...1) %>%
        mutate(key = as.numeric(key)) %>%
        mutate(value = value %>% str_replace(pattern = "'", replacement = "")) %>%
        split(.$column_name) %>%
        map(~ select(., -column_name)) %>%
        map(~ mutate(., value = as_factor(value)))

    for (i in seq_along(definitions_list)) {
        list_name <- names(definitions_list)[i]
        colnames(definitions_list[[i]]) <- c(list_name, paste0(list_name, "_value"))
    }

    data_merged_tbl <- list(HR_Data = data) %>%
        append(definitions_list, after = 1) %>%
        reduce(left_join) %>%
        select(-one_of(names(definitions_list))) %>%
        set_names(str_replace_all(names(.), pattern = "_value",
                                            replacement = "")) %>%
        select(sort(names(.))) %>%
        mutate_if(is.character, as.factor) %>%
        mutate(
            BusinessTravel = BusinessTravel %>% fct_relevel("Non-Travel",
                                                            "Travel_Rarely",
                                                            "Travel_Frequently"),
            MaritalStatus  = MaritalStatus %>% fct_relevel("Single",
                                                          "Married",
                                                          "Divorced")
```

```r
        )

    return(data_merged_tbl)

}
```

```r
employee_attrition_tbl <- read_csv("C:/Users/Lenovo/OneDrive/Desktop/daqtascience/ss24-bdml-Fahad221999/
definitions_raw_tbl    <- read_excel("C:/Users/Lenovo/OneDrive/Desktop/daqtascience/ss24-bdml-Fahad2219

employee_attrition_readable_tbl <- process_hr_data_readable(employee_attrition_tbl, definitions_raw_tbl)

# Split into test and train
set.seed(seed = 1113)
split_obj <- rsample::initial_split(employee_attrition_readable_tbl, prop = 0.85)

# Assign training and test data
train_readable_tbl <- training(split_obj)
test_readable_tbl  <- testing(split_obj)

recipe_obj <- recipe(Attrition ~ ., data = train_readable_tbl) %>%
                step_zv(all_predictors()) %>%
                step_mutate_at(c("JobLevel", "StockOptionLevel"), fn = as.factor) %>%
                prep()

train_tbl <- bake(recipe_obj, new_data = train_readable_tbl)
test_tbl  <- bake(recipe_obj, new_data = test_readable_tbl)

h2o.init()
```

```
##  Connection successful!
##
## R is connected to the H2O cluster:
##     H2O cluster uptime:         11 minutes 15 seconds
##     H2O cluster timezone:       Europe/Berlin
##     H2O data parsing timezone:  UTC
##     H2O cluster version:        3.44.0.3
##     H2O cluster version age:    6 months and 5 days
##     H2O cluster name:           H2O_started_from_R_Lenovo_drs368
##     H2O cluster total nodes:    1
##     H2O cluster total memory:   1.34 GB
##     H2O cluster total cores:    8
##     H2O cluster allowed cores:  8
##     H2O cluster healthy:        TRUE
##     H2O Connection ip:          localhost
##     H2O Connection port:        54321
##     H2O Connection proxy:       NA
##     H2O Internal Security:      FALSE
##     R Version:                  R version 4.3.0 (2023-04-21 ucrt)
```

```r
#automl_leader <- h2o.loadModel("C:/Users/Lenovo/OneDrive/Desktop/daqtascience/ss24-bdml-Fahad221999/St
split_h2o <- h2o.splitFrame(as.h2o(train_tbl), ratios = c(0.85), seed = 1234)
```

```
##   |                                                                              |
```

```r
train_h2o <- split_h2o[[1]]
valid_h2o <- split_h2o[[2]]
test_h2o  <- as.h2o(test_tbl)
```

```
##   |                                                                              |
```

```r
# Set the target and predictors
y <- "Attrition"
x <- setdiff(names(train_h2o), y)

automl_models_h2o <- h2o.automl(
  x = x,
  y = y,
  training_frame    = train_h2o,
  validation_frame  = valid_h2o,
  leaderboard_frame = test_h2o,
  max_runtime_secs  = 30,
  nfolds            = 5
)
```

```
##   |                                                                              |
## 23:49:26.526: User specified a validation frame with cross-validation still enabled. Please note tha
## 23:49:26.526: AutoML: XGBoost is not available; skipping it.  |
```

```r
automl_leader <- automl_models_h2o@leader
```

```r
explainer <- train_tbl %>%
    select(-Attrition) %>%
    lime(
        model           = automl_leader,
        bin_continuous  = TRUE,
        n_bins          = 4,
        quantile_bins   = TRUE
    )

explainer
```

```
## $model
## Model Details:
## ==============
##
## H2OBinomialModel: stackedensemble
## Model ID:  StackedEnsemble_AllModels_2_AutoML_3_20240625_234926
## Model Summary for Stacked Ensemble:
##                                          key            value
## 1                          Stacking strategy cross_validation
## 2       Number of base models (used / total)              7/9
## 3           # GBM base models (used / total)              4/5
## 4  # DeepLearning base models (used / total)              1/1
## 5           # GLM base models (used / total)              1/1
## 6           # DRF base models (used / total)              1/2
## 7                      Metalearner algorithm              GLM
```

```
## 8        Metalearner fold assignment scheme           Random
## 9                       Metalearner nfolds                5
## 10                  Metalearner fold_column               NA
## 11        Custom metalearner hyperparameters             None
##
##
## H2OBinomialMetrics: stackedensemble
## ** Reported on training data. **
##
## MSE:  0.05140328
## RMSE:  0.2267229
## LogLoss:  0.191248
## Mean Per-Class Error:  0.120203
## AUC:  0.9495546
## AUCPR:  0.8723406
## Gini:  0.8991093
##
## Confusion Matrix (vertical: actual; across: predicted) for F1-optimal threshold:
##         No Yes    Error      Rate
## No     884  25 0.027503   =25/909
## Yes     33 122 0.212903   =33/155
## Totals 917 147 0.054511  =58/1064
##
## Maximum Metrics: Maximum metrics at their respective thresholds
##                         metric threshold      value idx
## 1                       max f1  0.315089   0.807947 122
## 2                       max f2  0.223710   0.815085 161
## 3                 max f0point5  0.526217   0.871143  79
## 4                 max accuracy  0.335889   0.945489 117
## 5                max precision  0.977003   1.000000   0
## 6                   max recall  0.009379   1.000000 379
## 7              max specificity  0.977003   1.000000   0
## 8             max absolute_mcc  0.315089   0.776580 122
## 9   max min_per_class_accuracy  0.180499   0.883871 179
## 10 max mean_per_class_accuracy  0.223710   0.894854 161
## 11                     max tns  0.977003 909.000000   0
## 12                     max fns  0.977003 154.000000   0
## 13                     max fps  0.000543 909.000000 399
## 14                     max tps  0.009379 155.000000 379
## 15                     max tnr  0.977003   1.000000   0
## 16                     max fnr  0.977003   0.993548   0
## 17                     max fpr  0.000543   1.000000 399
## 18                     max tpr  0.009379   1.000000 379
##
## Gains/Lift Table: Extract with `h2o.gainsLift(<model>, <data>)` or `h2o.gainsLift(<model>, valid=<T/
## H2OBinomialMetrics: stackedensemble
## ** Reported on validation data. **
##
## MSE:  0.1031014
## RMSE:  0.3210941
## LogLoss:  0.3407438
## Mean Per-Class Error:  0.1825994
## AUC:  0.8634085
## AUCPR:  0.7141804
```

```
## Gini:  0.726817
##
## Confusion Matrix (vertical: actual; across: predicted) for F1-optimal threshold:
##         No Yes    Error     Rate
## No     132  15 0.102041  =15/147
## Yes     10  28 0.263158   =10/38
## Totals 142  43 0.135135  =25/185
##
## Maximum Metrics: Maximum metrics at their respective thresholds
##                         metric threshold      value idx
## 1                       max f1  0.275466   0.691358  42
## 2                       max f2  0.275466   0.717949  42
## 3                 max f0point5  0.484779   0.730769  22
## 4                 max accuracy  0.484779   0.875676  22
## 5                max precision  0.951513   1.000000   0
## 6                   max recall  0.016399   1.000000 146
## 7              max specificity  0.951513   1.000000   0
## 8             max absolute_mcc  0.275466   0.607169  42
## 9   max min_per_class_accuracy  0.146976   0.775510  62
## 10 max mean_per_class_accuracy  0.275466   0.817401  42
## 11                     max tns  0.951513 147.000000   0
## 12                     max fns  0.951513  37.000000   0
## 13                     max fps  0.001142 147.000000 184
## 14                     max tps  0.016399  38.000000 146
## 15                     max tnr  0.951513   1.000000   0
## 16                     max fnr  0.951513   0.973684   0
## 17                     max fpr  0.001142   1.000000 184
## 18                     max tpr  0.016399   1.000000 146
##
## Gains/Lift Table: Extract with ‘h2o.gainsLift(<model>, <data>)‘ or ‘h2o.gainsLift(<model>, valid=<T/
## H2OBinomialMetrics: stackedensemble
## ** Reported on cross-validation data. **
## ** 5-fold cross-validation on training data (Metrics computed for combined holdout predictions) **
##
## MSE:  0.08491412
## RMSE:  0.2914003
## LogLoss:  0.2998893
## Mean Per-Class Error:  0.2229249
## AUC:  0.8389403
## AUCPR:  0.6145444
## Gini:  0.6778807
##
## Confusion Matrix (vertical: actual; across: predicted) for F1-optimal threshold:
##         No Yes    Error       Rate
## No     838  71 0.078108    =71/909
## Yes     57  98 0.367742    =57/155
## Totals 895 169 0.120301  =128/1064
##
## Maximum Metrics: Maximum metrics at their respective thresholds
##                         metric threshold      value idx
## 1                       max f1  0.289824   0.604938 136
## 2                       max f2  0.221571   0.644970 170
## 3                 max f0point5  0.454259   0.650888  75
## 4                 max accuracy  0.454259   0.895677  75
```

```
## 5                max precision  0.968870   1.000000   0
## 6                   max recall  0.000571   1.000000 399
## 7              max specificity  0.968870   1.000000   0
## 8             max absolute_mcc  0.321175   0.537024 123
## 9   max min_per_class_accuracy  0.150630   0.774194 219
## 10 max mean_per_class_accuracy  0.221571   0.787807 170
## 11                     max tns  0.968870 909.000000   0
## 12                     max fns  0.968870 154.000000   0
## 13                     max fps  0.000571 909.000000 399
## 14                     max tps  0.000571 155.000000 399
## 15                     max tnr  0.968870   1.000000   0
## 16                     max fnr  0.968870   0.993548   0
## 17                     max fpr  0.000571   1.000000 399
## 18                     max tpr  0.000571   1.000000 399
##
## Gains/Lift Table: Extract with `h2o.gainsLift(<model>, <data>)` or `h2o.gainsLift(<model>, valid=<T/F
## Cross-Validation Metrics Summary:
##                mean       sd cv_1_valid cv_2_valid cv_3_valid cv_4_valid
## accuracy   0.895377 0.019105   0.909502   0.891753   0.901869   0.909910
## auc        0.844237 0.029927   0.891204   0.843706   0.849018   0.814055
## err        0.104623 0.019105   0.090498   0.108247   0.098131   0.090090
## err_count 22.200000 3.834058  20.000000  21.000000  21.000000  20.000000
## f0point5   0.637222 0.055689   0.684524   0.703125   0.590909   0.630631
##           cv_5_valid
## accuracy    0.863850
## auc         0.823201
## err         0.136150
## err_count  29.000000
## f0point5    0.576923
##
## ---
##                         mean        sd cv_1_valid cv_2_valid cv_3_valid
## precision           0.644412  0.050905   0.676471   0.692308   0.619048
## r2                  0.317661  0.075587   0.381526   0.406810   0.307976
## recall              0.620983  0.113304   0.718750   0.750000   0.500000
## residual_deviance 126.417040 12.348837 118.584816 132.572300 111.851640
## rmse                0.290019  0.020332   0.276741   0.299416   0.271777
## specificity         0.939603  0.022392   0.941799   0.924051   0.957447
##                   cv_4_valid cv_5_valid
## precision           0.666667   0.567568
## r2                  0.261196   0.230798
## recall              0.518518   0.617647
## residual_deviance 125.408960 143.667470
## rmse                0.280938   0.321223
## specificity         0.964103   0.910615
##
## $preprocess
## function (x)
## x
## <bytecode: 0x000001f16a923ad8>
## <environment: 0x000001f16a9184e0>
##
## $bin_continuous
## [1] TRUE
```

```
##
## $n_bins
## [1] 4
##
## $quantile_bins
## [1] TRUE
##
## $use_density
## [1] TRUE
##
## $feature_type
##                      Age           BusinessTravel                DailyRate
##                "numeric"                 "factor"                "numeric"
##               Department         DistanceFromHome                Education
##                 "factor"                "numeric"                 "factor"
##           EducationField           EmployeeNumber  EnvironmentSatisfaction
##                 "factor"                "numeric"                 "factor"
##                   Gender               HourlyRate           JobInvolvement
##                 "factor"                "numeric"                 "factor"
##                 JobLevel                  JobRole          JobSatisfaction
##                 "factor"                 "factor"                 "factor"
##            MaritalStatus            MonthlyIncome              MonthlyRate
##                 "factor"                "numeric"                "numeric"
##       NumCompaniesWorked                 OverTime        PercentSalaryHike
##                "numeric"                 "factor"                "numeric"
##        PerformanceRating RelationshipSatisfaction         StockOptionLevel
##                 "factor"                 "factor"                 "factor"
##        TotalWorkingYears    TrainingTimesLastYear          WorkLifeBalance
##                "numeric"                "numeric"                 "factor"
##           YearsAtCompany       YearsInCurrentRole  YearsSinceLastPromotion
##                "numeric"                "numeric"                "numeric"
##     YearsWithCurrManager
##                "numeric"
##
## $bin_cuts
## $bin_cuts$Age
##   0%  25%  50%  75% 100%
##   18   30   36   43   60
##
## $bin_cuts$BusinessTravel
## NULL
##
## $bin_cuts$DailyRate
##   0%  25%  50%  75% 100%
##  102  465  797 1147 1499
##
## $bin_cuts$Department
## NULL
##
## $bin_cuts$DistanceFromHome
##   0%  25%  50%  75% 100%
##    1    2    7   14   29
##
## $bin_cuts$Education
```

```
## NULL
##
## $bin_cuts$EducationField
## NULL
##
## $bin_cuts$EmployeeNumber
##   0%  25%  50%  75% 100%
##    1  511 1040 1573 2065
##
## $bin_cuts$EnvironmentSatisfaction
## NULL
##
## $bin_cuts$Gender
## NULL
##
## $bin_cuts$HourlyRate
##   0%  25%  50%  75% 100%
##   30   49   66   83  100
##
## $bin_cuts$JobInvolvement
## NULL
##
## $bin_cuts$JobLevel
## NULL
##
## $bin_cuts$JobRole
## NULL
##
## $bin_cuts$JobSatisfaction
## NULL
##
## $bin_cuts$MaritalStatus
## NULL
##
## $bin_cuts$MonthlyIncome
##    0%   25%   50%   75%  100%
##  1051  2929  4908  8474 19999
##
## $bin_cuts$MonthlyRate
##    0%   25%   50%   75%  100%
##  2094  8423 14470 20689 26968
##
## $bin_cuts$NumCompaniesWorked
##   0%  25%  50%  75% 100%
##    0    1    2    4    9
##
## $bin_cuts$OverTime
## NULL
##
## $bin_cuts$PercentSalaryHike
##   0%  25%  50%  75% 100%
##   11   12   14   18   25
##
## $bin_cuts$PerformanceRating
```

```
## NULL
##
## $bin_cuts$RelationshipSatisfaction
## NULL
##
## $bin_cuts$StockOptionLevel
## NULL
##
## $bin_cuts$TotalWorkingYears
##   0%  25%  50%  75% 100%
##    0    6   10   15   38
##
## $bin_cuts$TrainingTimesLastYear
##   0%  25%  50% 100%
##    0    2    3    6
##
## $bin_cuts$WorkLifeBalance
## NULL
##
## $bin_cuts$YearsAtCompany
##   0%  25%  50%  75% 100%
##    0    3    5    9   37
##
## $bin_cuts$YearsInCurrentRole
##   0%  25%  50%  75% 100%
##    0    2    3    7   18
##
## $bin_cuts$YearsSinceLastPromotion
##   0%  50%  75% 100%
##    0    1    2   15
##
## $bin_cuts$YearsWithCurrManager
##   0%  25%  50%  75% 100%
##    0    2    3    7   17
##
##
## $feature_distribution
## $feature_distribution$Age
##
##         1         2         3         4
## 0.2602082 0.2834267 0.2217774 0.2345877
##
## $feature_distribution$BusinessTravel
##
##        Non-Travel     Travel_Rarely Travel_Frequently
##         0.1000801         0.7181745         0.1817454
##
## $feature_distribution$DailyRate
##
##         1         2         3         4
## 0.2514011 0.2489992 0.2497998 0.2497998
##
## $feature_distribution$Department
##
```

```
##        Human Resources Research & Development                  Sales
##             0.04323459             0.65092074             0.30584468
##
## $feature_distribution$DistanceFromHome
##
##         1         2         3         4
## 0.2954363 0.2369896 0.2241793 0.2433947
##
## $feature_distribution$Education
##
## Below College       College      Bachelor        Master        Doctor
##    0.11689351    0.18895116    0.38510809    0.27461970    0.03442754
##
## $feature_distribution$EducationField
##
##  Human Resources    Life Sciences        Marketing          Medical
##       0.01761409       0.41793435       0.10888711       0.31144916
##            Other Technical Degree
##       0.05444355       0.08967174
##
## $feature_distribution$EmployeeNumber
##
##         1         2         3         4
## 0.2506005 0.2497998 0.2497998 0.2497998
##
## $feature_distribution$EnvironmentSatisfaction
##
##       Low    Medium      High Very High
## 0.1913531 0.1961569 0.3018415 0.3106485
##
## $feature_distribution$Gender
##
##    Female      Male
## 0.4123299 0.5876701
##
## $feature_distribution$HourlyRate
##
##         1         2         3         4
## 0.2618094 0.2473979 0.2449960 0.2457966
##
## $feature_distribution$JobInvolvement
##
##        Low     Medium       High  Very High
## 0.05684548 0.25780624 0.58927142 0.09607686
##
## $feature_distribution$JobLevel
##
##          1          2          3          4          5
## 0.36829464 0.36509207 0.14651721 0.07526021 0.04483587
##
## $feature_distribution$JobRole
##
## Healthcare Representative           Human Resources     Laboratory Technician
##                0.08646918                0.03682946                0.18174540
```

```
##                  Manager   Manufacturing Director        Research Director
##               0.06885508               0.09927942               0.05924740
##       Research Scientist          Sales Executive     Sales Representative
##               0.18654924               0.22337870               0.05764612
##
## $feature_distribution$JobSatisfaction
##
##       Low    Medium      High Very High
## 0.1873499 0.1985588 0.3018415 0.3122498
##
## $feature_distribution$MaritalStatus
##
##    Single   Married  Divorced
## 0.3306645 0.4571657 0.2121697
##
## $feature_distribution$MonthlyIncome
##
##         1         2         3         4
## 0.2506005 0.2497998 0.2497998 0.2497998
##
## $feature_distribution$MonthlyRate
##
##         1         2         3         4
## 0.2506005 0.2497998 0.2497998 0.2497998
##
## $feature_distribution$NumCompaniesWorked
##
##          1          2          3          4
## 0.48118495 0.09927942 0.20496397 0.21457166
##
## $feature_distribution$OverTime
##
##        No       Yes
## 0.7165733 0.2834267
##
## $feature_distribution$PercentSalaryHike
##
##         1         2         3         4
## 0.2866293 0.2738191 0.2289832 0.2105685
##
## $feature_distribution$PerformanceRating
##
##         Low        Good   Excellent Outstanding
##   0.0000000   0.0000000   0.8414732   0.1585268
##
## $feature_distribution$RelationshipSatisfaction
##
##       Low    Medium      High Very High
## 0.1889512 0.2161729 0.3018415 0.2930344
##
## $feature_distribution$StockOptionLevel
##
##          0          1          2          3
## 0.43554844 0.40592474 0.10168135 0.05684548
```

```
##
## $feature_distribution$TotalWorkingYears
##
##         1         2         3         4
## 0.3050440 0.3306645 0.1224980 0.2417934
##
## $feature_distribution$TrainingTimesLastYear
##
##         1         2         3
## 0.4603683 0.3306645 0.2089672
##
## $feature_distribution$WorkLifeBalance
##
##        Bad       Good     Better       Best
## 0.05204163 0.22497998 0.61889512 0.10408327
##
## $feature_distribution$YearsAtCompany
##
##         1         2         3         4
## 0.3226581 0.2137710 0.2217774 0.2417934
##
## $feature_distribution$YearsInCurrentRole
##
##          1          2          3          4
## 0.46757406 0.08726982 0.27542034 0.16973579
##
## $feature_distribution$YearsSinceLastPromotion
##
##         1         2         3
## 0.6413131 0.1120897 0.2465973
##
## $feature_distribution$YearsWithCurrManager
##
##          1          2          3          4
## 0.46357086 0.09767814 0.25300240 0.18574860
##
##
## attr(,"class")
## [1] "data_frame_explainer" "explainer"            "list"
```

```
explanation <- test_tbl %>%
    slice(1:20) %>%
    select(-Attrition) %>%
    lime::explain(

        # Pass our explainer object
        explainer = explainer,
        # Because it is a binary classification model: 1
        n_labels   = 1,
        # number of features to be returned
        n_features = 8,
        # number of localized linear models
        n_permutations = 5000,
        # Let's start with 1
```

```r
        kernel_width   = 1
    )
```

```
##   |                                                                              |
##   |                                                                              |
```

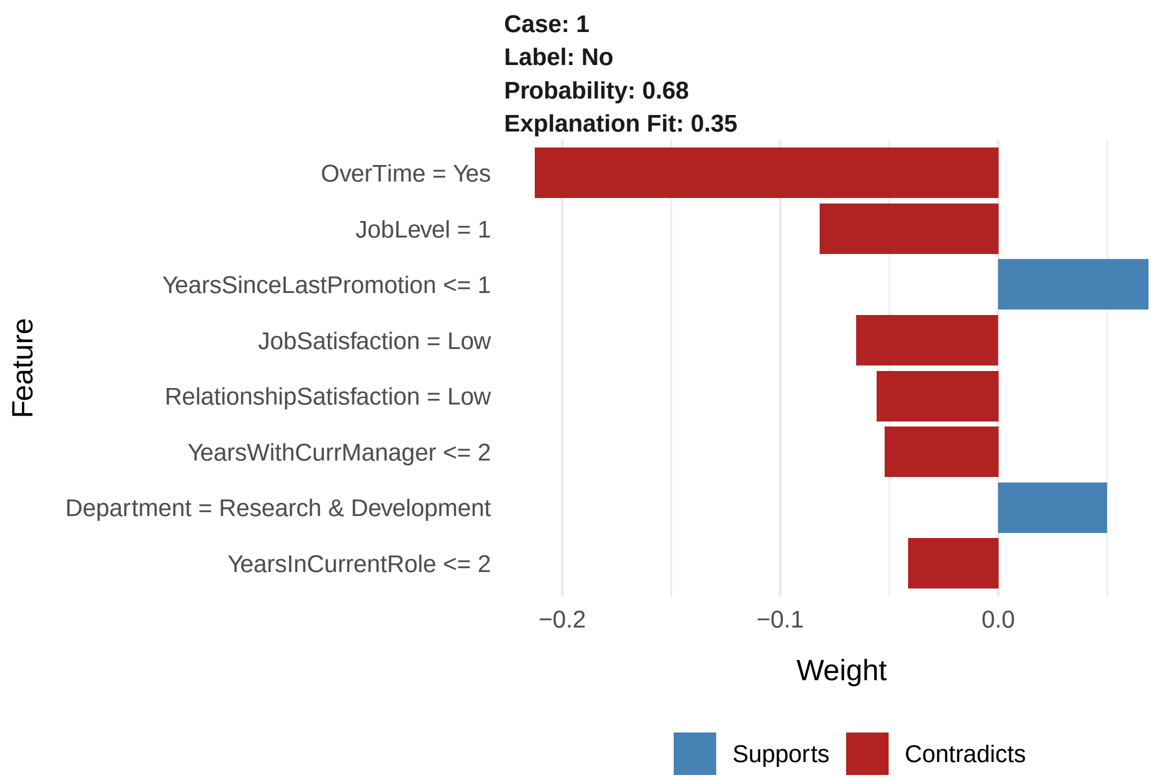
```r
explanation
```

```
## # A tibble: 160 x 13
##    model_type   case  label label_prob model_r2 model_intercept model_prediction
##    <chr>        <chr> <chr>      <dbl>    <dbl>           <dbl>            <dbl>
##  1 classificat~ 1     No         0.678    0.348           0.920            0.530
##  2 classificat~ 1     No         0.678    0.348           0.920            0.530
##  3 classificat~ 1     No         0.678    0.348           0.920            0.530
##  4 classificat~ 1     No         0.678    0.348           0.920            0.530
##  5 classificat~ 1     No         0.678    0.348           0.920            0.530
##  6 classificat~ 1     No         0.678    0.348           0.920            0.530
##  7 classificat~ 1     No         0.678    0.348           0.920            0.530
##  8 classificat~ 1     No         0.678    0.348           0.920            0.530
##  9 classificat~ 2     No         0.803    0.416           0.762            0.652
## 10 classificat~ 2     No         0.803    0.416           0.762            0.652
## # i 150 more rows
## # i 6 more variables: feature <chr>, feature_value <dbl>, feature_weight <dbl>,
## #   feature_desc <chr>, data <list>, prediction <list>
```

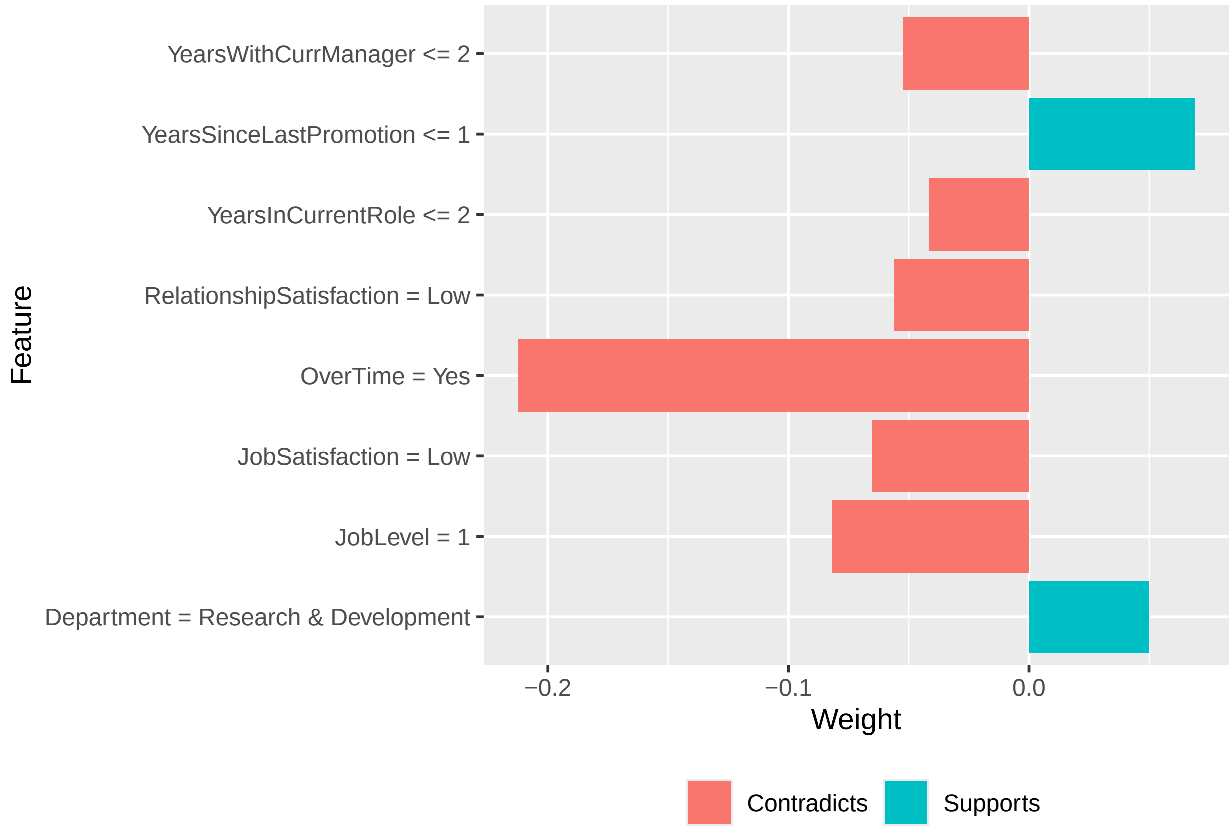
```r
explanation %>%
  as.tibble()
```

```
## # A tibble: 160 x 13
##    model_type   case  label label_prob model_r2 model_intercept model_prediction
##    <chr>        <chr> <chr>      <dbl>    <dbl>           <dbl>            <dbl>
##  1 classificat~ 1     No         0.678    0.348           0.920            0.530
##  2 classificat~ 1     No         0.678    0.348           0.920            0.530
##  3 classificat~ 1     No         0.678    0.348           0.920            0.530
##  4 classificat~ 1     No         0.678    0.348           0.920            0.530
##  5 classificat~ 1     No         0.678    0.348           0.920            0.530
##  6 classificat~ 1     No         0.678    0.348           0.920            0.530
##  7 classificat~ 1     No         0.678    0.348           0.920            0.530
##  8 classificat~ 1     No         0.678    0.348           0.920            0.530
##  9 classificat~ 2     No         0.803    0.416           0.762            0.652
## 10 classificat~ 2     No         0.803    0.416           0.762            0.652
## # i 150 more rows
## # i 6 more variables: feature <chr>, feature_value <dbl>, feature_weight <dbl>,
## #   feature_desc <chr>, data <list>, prediction <list>
```

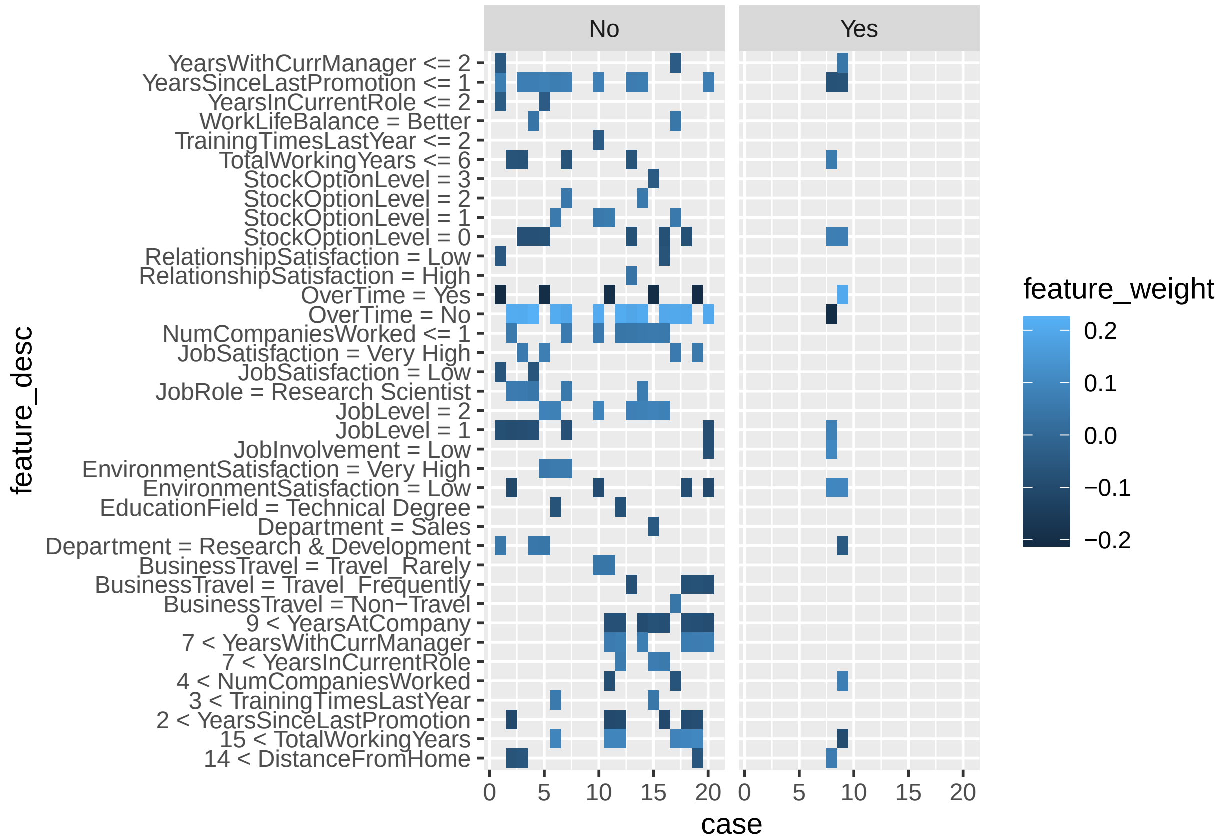
```r
case_1 <- explanation %>%
    filter(case == 1)

case_1 %>%
    plot_features()
```




```
case_1 %>%
  ggplot(aes(y=feature_desc, x =feature_weight)) +
  geom_col(aes(fill = feature_weight > 0)) +
  xlab("Weight") +
  ylab("Feature") +
  scale_fill_discrete(name = "", labels = c("Contradicts", "Supports")) +
  theme(legend.position = "bottom")
```




```
explanation %>%
  mutate(case = as.double(case)) %>%
  ggplot(aes(y=feature_desc, x =case, fill = feature_weight)) +
  geom_tile() +
  facet_wrap(~label)
```


No
Yes
YearsWithCurrManager <= 2
YearsSinceLastPromotion <= 1
YearsInCurrentRole <= 2
WorkLifeBalance = Better
TrainingTimesLastYear <= 2
TotalWorkingYears <= 6
StockOptionLevel = 3
StockOptionLevel = 2
StockOptionLevel = 1
StockOptionLevel = 0
RelationshipSatisfaction = Low
RelationshipSatisfaction = High
OverTime = Yes
OverTime = No
NumCompaniesWorked <= 1
JobSatisfaction = Very High
JobSatisfaction = Low
JobRole = Research Scientist
JobLevel = 2
JobLevel = 1
JobInvolvement = Low
EnvironmentSatisfaction = Very High
EnvironmentSatisfaction = Low
EducationField = Technical Degree
Department = Sales
Department = Research & Development
BusinessTravel = Travel_Rarely
BusinessTravel = Travel_Frequently
BusinessTravel = Non−Travel
9 < YearsAtCompany
7 < YearsWithCurrManager
7 < YearsInCurrentRole
4 < NumCompaniesWorked
3 < TrainingTimesLastYear
2 < YearsSinceLastPromotion
15 < TotalWorkingYears
14 < DistanceFromHome
feature_desc
0 5 10 15 20
0 5 10 15 20
case
feature_weight
0.2
0.1
0.0
−0.1
−0.2