Package ‘MLDataR’

March 8, 2022

Type    Package
Title   Collection of Machine Learning Datasets for Supervised Machine Learning
Version 0.1.3
Maintainer Gary Hutson <hutsons-hacks@outlook.com>
Description Contains a collection of datasets for working with machine learning tasks. It will contain datasets for supervised machine learning (Jiang 2020)<doi:10.1016/j.beth.2020.05.002> and will include datasets for classification and regression. The aim of this package is to use data generated around health and other domains.
License MIT + file LICENSE
Encoding UTF-8
LazyData true
BugReports https://github.com/StatsGary/MLDataR/issues
Imports ConfusionTableR, dplyr, parsnip, rsample, recipes, workflows, ranger, caret, varhandle, OddsPlotty, ggplot2
RoxygenNote 7.1.2
Suggests rmarkdown, knitr
VignetteBuilder knitr
Depends R (>= 2.10)
NeedsCompilation no
Author Gary Hutson [aut, cre] (<https://orcid.org/0000-0003-3534-6143>), Asif Laldin [aut], Isabella Velásquez [aut]
Repository CRAN
Date/Publication 2022-03-08 13:00:15 UTC
R topics documented:

care_home_incidents .................................................. 2
csgo ................................................................. 3
diabetes_data .......................................................... 4
heartdisease ............................................................ 5
long_stayers .............................................................. 6
PreDiabetes ............................................................... 8
thyroid_disease ......................................................... 9

Index 13

care_home_incidents Care Home Incidents

Description

dataset that has been synthetically generated against real data

Usage
care_home_incidents

Format

A data frame with 1216 rows and 12 variables:

CareHomeFail a binary indicator to specify whether a certain care home is failing
WeightLoss aggregation of incidents indicating weight loss in patient
Medication medication missed aggregation
Falls Recorded number of patient falls
Choking Number of patient choking incidents
UnexpectedDeaths unexpected deaths in the care home
Bruising Number of bruising incidents in the care home
Absconding Absconding from the care home setting
ResidentAbuseByResident Abuse conducted by one care home resident against another
ResidentAbuseByStaff Incidents of resident abuse by staff
ResidentAbuseOnStaff Incidents of residents abusing staff
Wounds Unexplained wounds against staff

Source

Collected by Gary Hutson <hutsons-hacks@outlook.com>, Jan-2022
Examples

```r
library(dplyr)
data(care_home_incidents)
# Convert diabetes data to factor
ch_incs <- care_home_incidents %>%
  mutate(CareHomeFail = as.factor(CareHomeFail))
ch_incs %>% glimpse()
# Check factor
factor(ch_incs$CareHomeFail)
```

### Description

csgo

### Usage

csgo

### Format

A data frame with 1,133 rows and 17 variables:

- **map**: Map on which the match was played
- **day**: Day of the month
- **month**: Month of the year
- **year**: Year
- **date**: Date of match DD/MM/YYYY
- **wait_time_s**: Time waited to find match
- **match_time_s**: Total match length in seconds
- **team_a_rounds**: Number of rounds played as Team A
- **team_b_rounds**: Number of rounds played as Team B
- **ping**: Maximum ping in milliseconds; the signal that’s sent from one computer to another on the same network
- **kills**: Number of kills accumulated in match; max 5 per round
- **assists**: Number of assists accumulated in a match, inflicting opponent with more than 50 percent damage, who is then killed by another player accumulated in match max 5 per round
- **deaths**: Number of times player died during match; max 1 per round
- **mvps**: Most Valuable Player award
- **hs_percent**: Percentage of kills that were a result from a shot to opponent’s head
- **points**: Number of points accumulated during match. Points are gained from kills, assists, bomb defuses & bomb plants. Points are lost for suicide and friendly kills
- **result**: The result of the match, Win, Loss, Draw
Source

Extracted by Asif Laldin <a.laldin@nhs.net>, March-2019

---

diabetes_data  Diabetes datasets

Description

Diabetes datasets

Usage

diabetes_data

Format

A data frame with 520 rows and 17 variables:

- **Age**  age of the patient presenting with diabetes
- **Gender**  gender of the patient with diabetes
- **ExcessUrination**  if the patient has a history of excessive urination
- **Polydipsia**  abnormal thirst, accompanied by the excessive intake of water or fluid
- **WeightLossSudden**  Sudden weight loss that has recently occurred
- **Fatigue**  Fatigue or weakness
- **Polyphagia**  excessive or extreme hunger
- **GenitalThrush**  patient has thrush fungus on or near their genital region
- **BlurredVision**  history of blurred vision
- **Itching**  skin itching
- **Irritability**  general irritability and mood issues
- **DelayHealing**  delayed healing of wounds
- **PartialPsoriasis**  partial psoriasis on the body
- **MuscleStiffness**  stiffness of the muscles
- **Alopecia**  scalp alopecia and hair shedding
- **Obesity**  Classified as obese
- **DiabeticClass**  Class label to indicate whether the patient is diabetic or not

Source

Collected by Gary Hutson <hutsons-hacks@outlook.com>, Dec-2021
heartdisease

Examples

```r
library(dplyr)
data(diabetes_data)
# Convert diabetes data to factor'
diabetes_data <- diabetes_data %>%
glimpse() %>%
mutate(DiabeticClass = as.factor(DiabeticClass))
is.factor(diabetes_data$DiabeticClass)
```

heartdisease

Heart disease dataset

Description

The dataset is to be used with a supervised classification ML model to classify heart disease.

Usage

heartdisease

Format

A data frame with 918 rows and 10 variables:

- **Age**: age of the patient presenting with heart disease
- **Sex**: gender of the patient
- **RestingBP**: blood pressure for resting heart beat
- **Cholesterol**: Cholesterol reading
- **FastingBS**: blood sample of glucose after a patient fasts [https://www.diabetes.co.uk/diabetes_care/fasting-blood-sugar-levels.html](https://www.diabetes.co.uk/diabetes_care/fasting-blood-sugar-levels.html)
- **RestingECG**: Resting echocardiography is an indicator of previous myocardial infarction e.g. heart attack
- **MaxHR**: Maximum heart rate
- **Angina**: chest pain caused by decreased flood flow [https://www.nhs.uk/conditions/angina/](https://www.nhs.uk/conditions/angina/)
- **HeartPeakReading**: reading at the peak of the heart rate
- **HeartDisease**: the classification label of whether patient has heart disease or not

Source

Collected by Gary Hutson <hutsons-hacks@outlook.com>, Dec-2021
Examples

```r
library(dplyr)
library(ConfusionTableR)
data(heartdisease)

# Convert diabetes data to factor
hd <- heartdisease %>%
glimpse() %>%
  mutate(HeartDisease = as.factor(HeartDisease))
# Check that the label is now a factor
is.factor(hd$HeartDisease)
# Dummy encoding
# Get categorical columns
hd_cat <- hd %>%
dplyr::select_if(is.character)
# Dummy encode the categorical variables
# Specify the columns to encode
cols <- c("RestingECG", "Angina", "Sex")
# Dummy encode using dummy_encoder in ConfusionTableR package
coded <- ConfusionTableR::dummy_encoder(hd_cat, cols, remove_original = TRUE)
coded <- coded %>%
  select(RestingECG_ST, RestingECG_LVH, Angina=Angina_Y, Sex=Sex_F)
# Remove column names we have encoded from original data frame
hd_one <- hd[, !names(hd) %in% cols]
# Bind the numerical data on to the categorical data
hd_final <- bind_cols(coded, hd_one)
# Output the final encoded data frame for the ML task
glimpse(hd_final)
```

long_stayers  Long stayers dataset

Description


Usage

long_stayers

Format

A data frame with 768 rows and 9 variables:

- **stranded.label**  binary classification label indicating whether stranded = 1 or not stranded=0
- **age**  age of the patient
**Care Home Referral** flag indicating whether referred from a private care home - 1=Care Home Referral and 0=Not a care home referral

**Medically Safe** flag indicating whether they are medically safe for discharge - 1=Medically safe and 0=Not medically safe

**HCOP** flag indicating health care for older person triage - 1=Yes triaged from HCOP and 0=Triaged from different department

**Mental Health Care** flag indicating whether they require mental health care - 1=MH assistance needed and 0=No history of mental health

**Periods of Previous Care** Count of the number of times they have been in hospital in last 12 months

**Admit Date** date the patient was admitted as an inpatient

**Frailty Index** indicates the type of frailty - nominal variable

**Source**
Prepared, acquired and adapted by Gary Hutson <hutsons-hacks@outlook.com>, Dec-2021. Synthetic data, based off live patient data from various NHS secondary health care trusts.

**Examples**

```r
library(dplyr)
library(ggplot2)
library(caret)
library(rsample)
library(varhandle)
data("long_stayers")
glimpse(long_stayers)
# Examine class imbalance
prop.table(table(long_stayers$stranded.label))
# Feature engineering
long_stayers <- long_stayers %>%
dplyr::mutate(stranded.label=factor(stranded.label)) %>%
dplyr::select(everything(), -c(admit_date))
# Feature encoding
cats <- select_if(long_stayers, is.character)
cat_dummy <- varhandle::to.dummy(cats$frailty_index, "frail_ind")
# Converts the frailty index column to dummy encoding and sets a column called "frail_ind" prefix
cat_dummy <- cat_dummy %>%
  as.data.frame() %>%
dplyr::select(-frail_ind.No_index_item) # Drop the field of interest
long_stayers <- long_stayers %>%
dplyr::select(-frailty_index) %>%
bind_cols(cat_dummy) %>% na.omit(.)
# Split the data
split <- rsample::initial_split(long_stayers, prop = 3/4)
train <- rsample::training(split)
test <- rsample::testing(split)
set.seed(123)
glm_class_mod <- caret::train(factor(stranded.label) ~ ., data = train,
  method = "glm")
```
print(glm_class_mod)
# Predict the probabilities
preds <- predict(glm_class_mod, newdata = test) # Predict class
pred_prob <- predict(glm_class_mod, newdata = test, type="prob") # Predict probs

predicted <- data.frame(preds, pred_prob)
test <- test %>%
  bind_cols(predicted) %>%
dplyr::rename(pred_class=preds)

library(ConfusionTableR)
cm <- ConfusionTableR::binary_class_cm(test$stranded.label, test$pred_class, positive="Stranded")
cm$record_level_cm

# Visualise odds ration
library(OddsPlotty)
plotty <- OddsPlotty::odds_plot(glm_class_mod$finalModel,
                                 title = "Odds Plot ",
                                 subtitle = "Showing odds of patient stranded",
                                 point_col = "#00f2ff",
                                 error_bar_colour = "black",
                                 point_size = .5,
                                 error_bar_width = .8,
                                 h_line_color = "red")

print(plotty)

---

**PreDiabetes**

**PreDiabetes dataset**

**Description**

PreDiabetes dataset

**Usage**

PreDiabetes

**Format**

A data frame with 3059 rows and 9 variables:

- **Age** age of the patient presenting with diabetes
- **Sex** sex of the patient with diabetes
- **IMD_Decile** Index of Multiple Deprivation Decile
- **BMI** Body Mass Index of patient
- **Age_PreDiabetes** age at pre diabetes diagnosis
- **HbA1C** average blood glucose mmol/mol
- **Time_Pre_To_Diabetes** time in years between pre-diabetes and diabetes diagnosis
- **Age_Diabetes** age at diabetes diagnosis
- **PreDiabetes_Checks_Before_Diabetes** number of pre-diabetes related primary care appointments before diabetes diagnosis
thyroid_disease

Source

Generated by Asif Laldin <a.laldin@nhs.net>, Jan-2022

Examples

```r
library(dplyr)
data(PreDiabetes)
# Convert diabetes data to factor'
diabetes_data <- PreDiabetes %>%
glimpse()
```

thyroid_disease  Thyroid disease dataset

Description

The dataset is to be used with a supervised classification ML model to classify thyroid disease. The dataset was sourced and adapted from the UCI Machine Learning repository [https://archive.ics.uci.edu/ml/index.php](https://archive.ics.uci.edu/ml/index.php).

Usage

thyroid_disease

Format

A data frame with 3772 rows and 28 variables:

- **ThyroidClass**: binary classification label indicating whether sick = 1 or negative=0
- **patient_age**: age of the patient
- **patient_gender**: flag indicating gender of patient - 1=Female and 0=Male
- **presc_thyroxine**: flag to indicate whether thyroxine replacement prescribed 1=Thyroxine prescribed
- **queried_why_on_thyroxine**: flag to indicate query has been actioned
- **presc_anthyroid_meds**: flag to indicate whether anti-thyroid medicine has been prescribed
- **sick**: flag to indicate sickness due to thyroxine depletion or over activity
- **pregnant**: flag to indicate whether the patient is pregnant
- **thyroid_surgery**: flag to indicate whether the patient has had thyroid surgery
- **radioactive_iodine_therapyI131**: indicates whether patient has had radioactive iodine treatment: [https://www.nhs.uk/conditions/thyroid-cancer/treatment/](https://www.nhs.uk/conditions/thyroid-cancer/treatment/)
- **query_hypothyroid**: flag to indicate under active thyroid query [https://www.nhs.uk/conditions/underactive-thyroid-hypothyroidism/](https://www.nhs.uk/conditions/underactive-thyroid-hypothyroidism/)
- **query_hyperthyroid**: flag to indicate over active thyroid query [https://www.nhs.uk/conditions/overactive-thyroid-hyperthyroidism/](https://www.nhs.uk/conditions/overactive-thyroid-hyperthyroidism/)
lithium  Lithium carbonate administered to decrease the level of thyroid hormones

goitre  flag to indicate swelling of the thyroid gland https://www.nhs.uk/conditions/goitre/

tumor  flag to indicate a tumor

hypopituitarism  flag to indicate a diagnosed under active thyroid

psych_condition  indicates whether a patient has a psychological condition

TSH_measured  a TSH level lower than normal indicates there is usually more than enough thyroid hormone in the body and may indicate hyperthyroidism

TSH_reading  the reading result of the TSH blood test

T3_measured  linked to TSH reading - when free triiodothyronine rise above normal this indicates hyperthyroidism

T3_reading  the reading result of the T3 blood test looking for above normal levels of free triiodothyronine

T4_measured  free thyroxine, also known as T4, is used with T3 and TSH tests to diagnose hyperthyroidism

T4_reading  the reading result of the T4 test

thyrox_util_rate_T4U_measured  flag indicating the thyroxine utilisation rate https://pubmed.ncbi.nlm.nih.gov/1685967/

thyrox_util_rate_T4U_reading  the result of the test

FTI_measured  flag to indicate measurement on the Free Thyroxine Index (FTI) https://endocrinology.testcatalog.org/show/FRTUP

FTI_reading  the result of the test mentioned above

ref_src [nominal] indicating the referral source of the patient

Source

Prepared and adapted by Gary Hutson <hutsons-hacks@outlook.com>, Dec-2021 and sourced from Garavan Institute and J. Ross Quinlan.

References

Thyroid disease records supplied by the Garavan Institute and J. Ross Quinlan.

Examples

library(dplyr)
library(ConfusionTableR)
library(parsnip)
library(rsample)
library(recipes)
library(ranger)
library(workflows)
data("thyroid_disease")
td <- thyroid_disease
# Create a factor of the class label to use in ML model
td$ThryroidClass <- as.factor(td$ThryroidClass)
# Check the structure of the data to make sure factor has been created
str(td)

# Remove missing values, or choose more advanced imputation option
td <- td[complete.cases(td),]

# Drop the column for referral source
td <- td %>%
dplyr::select(-ref_src)

# Analyse class imbalance
class_imbalance <- prop.table(table(td$ThryroidClass))
class_imbalance

# Divide the data into a training test split
set.seed(123)
split <- rsample::initial_split(td, prop=3/4)
train_data <- rsample::training(split)
test_data <- rsample::testing(split)

# Create recipe to upsample and normalise
set.seed(123)
td_recipe <-
  recipe(ThryroidClass ~ ., data=train_data) %>%
  step_normalize(all_predictors()) %>%
  step_zv(all_predictors())

# Instantiate the model
set.seed(123)
rf_mod <-
  parsnip::rand_forest() %>%
  set_engine("ranger") %>%
  set_mode("classification")

# Create the model workflow
set.seed(123)
td_wf <-
  workflow() %>%
  workflows::add_model(rf_mod) %>%
  workflows::add_recipe(td_recipe)

# Fit the workflow to our training data
set.seed(123)
td_rf_fit <-
  td_wf %>%
  fit(data = train_data)

# Extract the fitted data

# Predict the test set on the training set to see model performance
class_pred <- predict(td_rf_fit, test_data)
td_preds <- test_data %>%
  bind_cols(class_pred)

# Convert both to factors
td_preds$pred_class <- as.factor(td_preds$pred_class)
td_preds$ThryroidClass <- as.factor(td_preds$ThryroidClass)

# Evaluate the data with ConfusionTableR

# View Confusion matrix

cm$confusion_matrix
#View record level
cm$record_level_cm
Index

* CounterStrike
  csgo, 3
* Global
  csgo, 3
* NHS
  long_stayers, 6
* Offensive
  csgo, 3
* care
  care_home_incidents, 2
* classification
  care_home_incidents, 2
diabetes_data, 4
heartdisease, 5
* diabetes
  diabetes_data, 4
* disease
  heartdisease, 5
  thyroid_disease, 9
* eSports
  csgo, 3
* heart
  heartdisease, 5
* home
  care_home_incidents, 2
* incidents
  care_home_incidents, 2
* learning
  care_home_incidents, 2
diabetes_data, 4
heartdisease, 5
PreDiabetes, 8
* long
  long_stayers, 6
* machine
  care_home_incidents, 2
diabetes_data, 4
heartdisease, 5
PreDiabetes, 8

* patient
  long_stayers, 6
* prediabetes
  PreDiabetes, 8
* regression
  PreDiabetes, 8
* stay
  long_stayers, 6
* stranded
  long_stayers, 6
* supervised
  care_home_incidents, 2
* thyroid
  thyroid_disease, 9
care_home_incidents, 2
csgo, 3
diabetes_data, 4
heartdisease, 5
long_stayers, 6
PreDiabetes, 8
thyroid_disease, 9