Package ‘mdsr’

January 6, 2023

Title Complement to ‘Modern Data Science with R’

Version 0.2.7

Description A complement to *Modern Data Science with R*, both the first and second editions (ISBN: 978-0367191498, publisher URL: <https://www.routledge.com/Modern-Data-Science-with-R/Baumer-Kaplan-Horton/p/book/9780367191498>). This package contains data and code to complete exercises and reproduce examples from the text. It also facilitates connections to the SQL database server used in the book. Both editions of the book are supported by this package.

Depends R (>= 3.5.0)

License CC0

LazyData true

LazyDataCompression xz

Imports babynames, DBI, dbplyr, downloader, dplyr, fs, ggplot2, htmlwidgets, kableExtra, RMariaDB, skimr, stringr, tibble, webshot2

Suggests knitr, Lahman, leaflet, etl, macleish, mosaic, mosaicData, lubridate, sf, testthat, utf8

RoxygenNote 7.2.3

Encoding UTF-8


NeedsCompilation no

Author Benjamin S. Baumer [aut, cre] (<https://orcid.org/0000-0002-3279-0516>), Nicholas Horton [aut] (<https://orcid.org/0000-0003-3332-4311>), Daniel Kaplan [aut]

Maintainer Benjamin S. Baumer <ben.baumer@gmail.com>

Repository CRAN

Date/Publication 2023-01-06 15:50:13 UTC
R topics documented:

Cherry ................................................. 2
CholeraDeaths ........................................ 3
CIACountries ......................................... 4
DataSciencePapers ................................. 5
Elections ............................................... 6
Emails_train .......................................... 7
etl_NCI60 ........................................... 8
Headlines_train ..................................... 8
Macbeth_raw ........................................ 9
macros .................................................. 9
make_babynames_dist .............................. 11
mdsr_table ........................................... 11
MedicareCharges .................................... 12
MedicareProviders ................................. 13
Minneapolis2013 .................................... 14
MLB_teams ........................................... 15
NCI60_tiny .......................................... 16
ordway_birds ........................................ 17
Rnw2Rmd ............................................ 18
saratoga_houses .................................... 19
SAT_2010 ........................................... 20
save_webshot ........................................ 21
skim ..................................................... 22
src_scidb ........................................... 22
theme_mdsr ......................................... 23
Violations ............................................ 24
Votes .................................................. 25
world_cities ......................................... 26

Index 28

Cherry

Cherry Blossom runs

Description

Cherry Blossom runs

Usage

Cherry
Format

An object of class tbl_df (inherits from tbl, data.frame) with 41,248 rows and 8 columns. Each row refers to an individual runner in one race of the Cherry Blossom Ten Miler. The data cover the years 1999 to 2008. All of the runners listed ran at least two of the races in that period, some ran many more than that.

name.yob a unique identifier for each runner composed of the runner’s full name and year of birth.
age integer giving the runner’s age in the race whose result is being reported.
gun the number of minutes elapsed from the starter’s gun to the person crossing the finish line
net the number of minutes elapsed from the runner’s crossing the start line to crossing the finish line.
sex the runner’s sex
year the year of that race
previous integer specifying how many times previous to this race the runner had participated in the years 1999 to 2008.
nruns integer giving the total number of times that runner participated in the years from 1999 to 2008. The smallest is 2, the largest is 10.
nruns integer giving the total number of times that runner participated in the years from 1999 to 2008. The smallest is 2, the largest is 10.

Details

The Cherry Blossom 10 Mile Run is a road race held in Washington, D.C. in April each year. (The name comes from the famous cherry trees that are in bloom in April in Washington.) The results of this race are published at https://www.cherryblossom.org/post-race/race-results/.

Examples

```r
if (require(dplyr)) {
    Cherry %>%
    group_by(name.yob) %>%
    count() %>%
    group_by(n) %>%
    count(name = "appearances")
}
```

Description

Deaths and Pumps from 1854 London cholera outbreak

Deaths and Pumps from 1854 London cholera outbreak
CIACountries

Usage

CholeraDeaths

CholeraPumps

Format

An object of class `sf` whose data attribute has 250 rows and 2 columns.
An object of class `sf`.

Details

Both spatial objects are projected in EPSG:27700, aka the British National Grid.

Source


Examples

```r
if (require(sf)) {
  plot(st_geometry(CholeraDeaths))
}
```

CIACountries

Several variables on countries from the CIA Factbook, 2014.

Description

The CIA Factbook has geographic, demographic, and economic data on a country-by-country basis. In the description of the variables, the 4-digit number indicates the code used to specify that variable on the data and documentation web site.

Usage

CIACountries

Format

A data frame with the following variables for each of the Countries in the World. (236 countries are given.)

- **country**  Name of the country
- **pop** number of people, 2119
- **area** area (sq km), 2147
- **oil_prod** Crude oil - production (bbl/day), 2241
- **gdp** Gross Domestic Product per capita ($/person), 2001
DataSciencePapers

Source
From the CIA World Factbook, https://www.cia.gov/the-world-factbook/

References
https://github.com/factbook/factbook/blob/master/CATEGORIES.md

See Also
CIAdata

Examples
str(CIACountries)

DataSciencePapers  Data Science Papers from arXiv.org

Description
Papers matching the search string "Data Science" on arXiv.org in August, 2020

Usage
DataSciencePapers

Format
A data frame with 1089 observations on the following 15 variables.

id  unique arXiv.org identifier for the paper
submitted  date submitted
updated  date last updated
title  title of the paper
abstract  contents of the abstract
authors  authors of the paper
affiliations  affiliations of the authors
link_abstract  direct link to the abstract
link_pdf  direct link to the pdf
<table>
<thead>
<tr>
<th>Ward</th>
<th>Name of the country</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precinct</td>
<td>number of people, 2119</td>
</tr>
<tr>
<td>Registered.Voters.at.7am</td>
<td>area (sq km), 2147</td>
</tr>
<tr>
<td>Voters.Registering.at.Polls</td>
<td>Crude oil - production (bbl/day), 2241</td>
</tr>
<tr>
<td>gdp</td>
<td>Gross Domestic Product per capita ($/person), 2001</td>
</tr>
<tr>
<td>educ</td>
<td>education spending (% of GDP), 2206</td>
</tr>
<tr>
<td>roadways</td>
<td>Roadways per unit area (km/sq km), 2085</td>
</tr>
<tr>
<td>net_users</td>
<td>Fraction of Internet users (% of population), 2153</td>
</tr>
</tbody>
</table>
Description

The training dataset includes a set of email subject lines used for classification of whether the message is spam (unsolicited commercial content) or not. Many subject lines include subject matter inappropriate for classroom use. Given the volume of headlines containing such language (especially for spam == TRUE), user discretion is advised. This dataset is a random sample of 80% of the emails data.

The testing dataset is a random sample of 20% of the emails data.

Usage

Emails_train

Emails_test

Format

A data frame with 5,526 rows and 3 variables:

ids an integer vector
subjectline a character vector
type a character vector

A data frame with 1,382 rows and 3 variables:

Source

Originally retrieved from http://www.rdatasciencecases.org/Data.html

Examples

nrow(Emails_train)
nrow(Emails_test)
etl_NCI60

Load the NCI60 data from GitHub

Description

Load the NCI60 data from GitHub

Usage

etl_NCI60()

Examples

## Not run:
NCI60 <- etl_NCI60()

## End(Not run)

Headlines_train

This data comes from Chakraborty et. al., which combines headlines from a variety of news and clickbait sources. Some headlines contain subject matter inappropriate for classroom use. Given the volume of headlines containing such language (especially for clickbait == TRUE), this filtering might not catch all problematic headlines. User discretion is advised. The training dataset is a random sample of approximately 80% of the observations from the original dataset. The testing dataset is a random sample of the remaining 20% of the observations not found in the training set.

Usage

Headlines_train

Headlines_test

Format

A data frame with 18,360 rows and 3 variables:

- **title** a character vector
- **clickbait** a logical vector
- ids an integer vector

A data frame with 4,589 rows and 3 variables:
Macbeth_raw

Source

https://github.com/bhargaviparanjape/clickbait/

References

doi:10.1109/ASONAM.2016.7752207

Examples

nrow(Headlines_train)
nrow(Headlines_test)

<table>
<thead>
<tr>
<th>Macbeth_raw</th>
<th>Text of Macbeth</th>
</tr>
</thead>
</table>

Description

The entire text of Macbeth, stored in a character vector of length 1.

Usage

Macbeth_raw

Format

A character vector of length 1

Source

Project Gutenberg, https://www.gutenberg.org/ebooks/1129/

<table>
<thead>
<tr>
<th>macros</th>
<th>Replacements for LaTeX macros</th>
</tr>
</thead>
</table>

Description

Replacements for LaTeX macros
Usage

func(x, ...)

sql_func(x)

sql_word(x)

argument(x)

variable(x)

pkg(x, ...)

mdsr_data(x)

mdsr_person(x, ...)

vocab(x, ...)

index_entry(
    index_label = "subject",
    x,
    emph = FALSE,
    index = TRUE,
    .f = NULL,
    alt = NULL
)

Arguments

x text to wrap in macro

... arguments passed to index_entry

index_label the name of the index

emph Display the LaTeX entry in italics

index add LaTeX indexing?

.f function to apply to x during indexing

alt alternate character string to use for indexing

Examples

func("mutate")
func("mutate", index = FALSE)
func("left_join")
pkg("dplyr")
mdsr_person("Ben Baumer")
mdsr_person("Ben Baumer", emph = TRUE)
mdsr_person("Richard De Veaux")
make_babynames_dist

Wrangle babynames data

Description

Wrangle babynames data

Usage

make_babynames_dist()

Value

A tbl_df similar to babynames with a column for the estimated number of people alive in 2014.

Examples

BabynameDist <- make_babynames_dist()
if (require(dplyr)) {
  BabynameDist %>%
    filter(name == "Benjamin")
}

mdsr_table

Custom table output

Description

Custom table output

Usage

mdsr_table(x, ...)

mdsr_sql_explain_table(x, ...)

mdsr_sql_keys_table(x, ...)
Arguments

x    A data.frame
    ...
Arguments passed to kbl

Examples

mdsr_table(faithful)

---

MedicareCharges  Charges to and Payments from Medicare

Description

These data for 2011, released in May 2013, describe how much hospitals charged Medicare for various inpatient procedures, how many were performed, and how much Medicare actually paid.

Usage

MedicareCharges

Format

A data frame with 5,025 observations on the following 4 variables.

- **drg**: Code for the Diagnosis Related Group: a character string that looks like a number.
- **stateProvider**: the state providing the care.
- **num_charges**: the total number of charges.
- **mean_charge**: the average charge for each drg across each state

Details

These data are part of a set with DiagnosisRelatedGroup, which gives a description of the medical procedure associated with each DRG, and MedicareProviders, which translates idProvider into a name, address, state, Zip, etc..

These data have been pre-aggregated by state.

Source

Data from the Centers for Medicare and Medicaid Services. See https://data.cms.gov/provider-summary-by-type-of
medicare-inpatient-hospitals/

See Also

MedicareProviders
Examples

```r
data(MedicareCharges)
```

---

**MedicareProviders**  
*Medicare Providers*

**Description**

Name and location data for the medicare providers in the `MedicareCharges` data table.

**Usage**

`MedicareProviders`

**Format**

A data frame with 3337 observations on the following 7 variables.

- `idProvider`  
a unique number assigned to each provider
- `nameProvider`  
Name of the provider. (text string)
- `addressProvider`  
Street address of the provider. (text string)
- `cityProvider`  
The name of the city in which the provider is located. (factor)
- `stateProvider`  
The two-letter postal code of the state in which the provider is located. (factor)
- `zipProvider`  
The provider’s ZIP code. (factor)
- `referralRegion`  
An identifier for the region serviced by the provider.

**Details**

This data table is related to `MedicareCharges` data.

**Source**


**See Also**

`MedicareCharges`

**Examples**

```r
data(MedicareProviders)
```
Ballots in the 2013 Mayoral election in Minneapolis

Description

The choices marked on each (valid) ballot for the election, which was run using a rank-choice, instant runoff system.

Usage

Minneapolis2013

Format

A data frame with 80,101 observations on the following 5 variables. All are stored as character strings.

- **Precinct** Precincts are sub-divisions within Wards
- **First** The voter’s first choice
- **Second** The voter’s second choice
- **Third** The voter’s third choice
- **Ward** The city is divided spatially into districts or ‘wards’. These are further subdivided into precincts.

Details

Ballot information for the 2013 Minneapolis Mayoral election, which was run as a rank-choice election. In rank-choice, a voter can indicate first, second, and third choices. If a voter’s first choice is eliminated (by being last in the count across voters), the second choice is promoted to that voter’s first choice, and similarly third -> second. Eliminations are done successively until one candidate has a majority of the first-choice votes.

Source

Ballot data from the Minneapolis city government: [https://vote.minneapolismn.gov/results-data/election-results/2013/mayor/](https://vote.minneapolismn.gov/results-data/election-results/2013/mayor/)

References

Description of ranked-choice voting: [https://vote.minneapolismn.gov/ranked-choice-voting/](https://vote.minneapolismn.gov/ranked-choice-voting/)


**MLB_teams**

### Examples

```r
data(Minneapolis2013)
```

---

**MLB_teams**

*Data about recent major league baseball teams*

### Description

A dataset containing information about Major League Baseball teams from 2008-2014.

### Usage

`MLB_teams`

### Format

A `tbl_df` object.

- **yearID**: season in which the team played
- **teamID**: the team’s three character identifier
- **lgID**: the league in which the team played
- **W**: number of wins
- **L**: number of losses
- **WPct**: winning percentage
- **attendance**: number of fans in attendance
- **normAttend**: number of fans in attendance, relative to the team with the highest attendance in this sample (the 2008 New York Yankees)
- **payroll**: the sum of the salaries of the players on each team. Note that this number is only an estimate of the actual team payroll – and may not even be a very good one. Salaries are accumulated from `Salaries`
- **metroPop**: the size of the team’s home city’s metropolitan population, according to Wikipedia and the 2010 US Census
- **name**: the full name of the team

### Source


### See Also

`Teams`
NCI60_tiny  Gene expression in cancer

Description
The data come from a National Cancer Institute study of gene expression in cell lines drawn from various sorts of cancer.

Usage
NCI60_tiny

Cancer

Format
The expression data, NCI60_tiny is a dataframe of 41,078 gene probes (rows) and 60 cell lines (columns). The first column, Probe gives the name of the Agilent microarray probe. Each of the remaining columns is named for a cell line. The value is the log-2 expression associated with that probe for the cell line.

Probe  the name of the Agilent microarray probe

For Cancer:

otherCellLine  a character vector giving the name of one cell line
cellLine  a character vector giving the name of another cell line
correlation  the correlation between the two cell lines. See cor

An object of class tbl_df (inherits from tbl, data.frame) with 1770 rows and 3 columns.

Details
Cancer gives information about each cell line.

References
Staunton et al. (2001), PNAS (doi:10.1073/pnas.191368598)
CellMiner

See Also
Cancer

Examples
data(NCI60_tiny)
ordway_birds

ordway_birds  Birds captured and released at Ordway, complete and unclean

Description
The historical record of birds captured and released at the Katharine Ordway Natural History Study Area, a 278-acre preserve in Inver Grove Heights, Minnesota, owned and managed by Macalester College.

Usage
ordway_birds

Format
A data frame with 15,829 observations on the bird’s species, size, date found, and band number.

timestamp  a character vector
Year  a character vector
Day  a character vector
Month  a character vector
CaptureTime  a character vector
SpeciesName  a character vector
Sex  a character vector
Age  a character vector
BandNumber  a character vector
TrapID  a character vector
Weather  a character vector
BandingReport  a character vector
RecaptureYN  a character vector
RecaptureMonth  a character vector
RecaptureDay  a character vector
Condition  a character vector
Release  a character vector
Comments  a character vector
DataEntryPerson  a character vector
Weight  a character vector
WingChord  a character vector
Temperature a character vector
RecaptureOriginal a character vector
RecapturePrevious a character vector
TailLength a character vector

Timestamp indicates when the data were entered into an electronic record, not anything about the bird being described.

Details

There are many extraneous levels of variables such as species. Part of the purpose of this data set is to teach about data cleaning.

Source

Jerald Dosch, Dept. of Biology, Macalester College: the manager of the Study Area.

References

https://www.macalester.edu/ordway/

Examples

ordway_birds

---

Rnw2Rmd

Convert Rnw to Rmd

Description

Convert Rnw to Rmd

Usage

Rnw2Rmd(path, new_path = NULL)

Arguments

path A character vector of one or more paths.
new_path New file path. If new_path is existing directory, the file will be moved into that directory; otherwise it will be moved/renamed to the full path. Should either be the same length as path, or a single directory.
saratoga_houses  Saratoga Houses

Description
Saratoga Houses

Usage
saratoga_houses
saratoga_codes

Format
A tibble with 1728 rows and 16 variables:

price, lot_size, waterfront, age, land_value, construction, air_cond, fuel, heat, sewer, living_area, pct_college, bedrooms, fireplaces, bathrooms, rooms

@examples saratoga_houses
An object of class spec_tbl_df (inherits from tbl_df, tbl, data.frame) with 13 rows and 3 columns.
SAT_2010

State SAT scores from 2010

Description

SAT results by state for 2010

Usage

SAT_2010

Format

A data.frame with 50 rows and 9 variables.

- state: a factor with levels for each state
- expenditure: average expenditure per student (in each state)
- pupil_teacher_ratio: pupil to teacher ratio in that state
- salary: teacher salary (in 2010 US $)
- read: state average Reading SAT score
- math: state average Math SAT score
- write: state average Writing SAT score
- total: state average Total SAT score
- sat_pct: percent of students taking SAT in that state

Details

See also the earlier SAT dataset.

See Also

SAT
save_webshot

**Description**
Embedded webshot of leaflet map

**Usage**
```r
save_webshot(
  map,
  path_to_img,
  overwrite = FALSE,
  vwidth = 800,
  vheight = 600,
  cliprect = "viewport",
  ...
)
```

**Arguments**
- `map` A leaflet map object
- `path_to_img` A path to the image file to save
- `overwrite` Do you want to clobber any existing file?
- `vwidth` see `webshot`
- `vheight` see `webshot`
- `cliprect` see `webshot`
- `...` arguments passed to `webshot`

**Value**
a path to a PNG file

**Examples**
```r
## Not run:
if (require(leaflet)) {
  map <- leaflet() %>%
    addTiles() %>%
    addMarkers(lng = 174.768, lat = -36.852, popup = "The birthplace of R")
  save_webshot(map, tempfile())
}
## End(Not run)
```
**skim**

*Custom skimmer*

---

**Description**

Custom skimmer

**Usage**

```r
skim(data, ...)
```

**Arguments**

- `data` A tibble, or an object that can be coerced into a tibble.
- `...` Columns to select for skimming. When none are provided, the default is to skim all columns.

**Examples**

```r
skim(faithful)
```

---

**src_scidb**

*src_scidb*

---

**Description**

Connect to the scidb server on Amazon Web Services.

**Usage**

```r
src_scidb(dbname, ...)
```

```r
dbConnect_scidb(dbname, ...)
```

```r
mysql_scidb(dbname, ...)
```

**Arguments**

- `dbname` the name of the database to which you want to connect
- `...` arguments passed to *src_dbi* or *dbConnect*

**Details**

This is a public, read-only account. Any abuse will be considered a hostile act.
Value

For `src_scidb`, a `src_dbi` object
For `dbConnect_scidb`, a `MariaDBConnection-class` object
For `mysql_scidb`, a character vector of length 1 to be used as an `engine.ops` argument, or on the command line.

See Also

`src_dbi`
`MariaDBConnection-class`
`opts_chunk`

Examples

```r
## Not run:
db_air <- src_scidb("airlines")
db_air
## End(Not run)
## Not run:
db_air <- dbConnect_scidb("airlines")
db_air
if (require(DBI)) {
  dbListTables(db_air)
}
## End(Not run)

if (require(knitr)) {
  opts_chunk$set(engine.opts = mysql_scidb("airlines"))
}
```

---

`theme_mdsr`  

*MDSR themes*

Description

Graphical themes used in MDSR book

Usage

`theme_mdsr(base_size = 12, base_family = "Bookman")`

Arguments

- `base_size` base font size, given in pts.
- `base_family` base font family
Examples

```r
if (require(ggplot2)) {
  p <- ggplot(mtcars, aes(x = hp, y = mpg, color = factor(cyl))) +
             geom_point() + facet_wrap(~ am) + geom_smooth()
  p + theme_grey()
  p + theme_mdsr()
}
```

### Description

NYC Restaurant Health Violations

### Usage

Violations

ViolationCodes

Cuisines

### Format

A data frame with 480,621 observations on the following 16 variables.

- `camis` unique identifier
- `dba` full name doing business as
- `boro` borough of New York
- `building` building name
- `street` street address
- `zipcode` zipcode
- `phone` phone number
- `inspection_date` inspection date
- `action` action taken
- `violation_code` violation code, see `ViolationCodes`
- `score` inspection score
- `grade` inspection grade
- `grade_date` grade date
- `record_date` recording date
- `inspection_type` inspect type
- `cuisine_code` cuisine code, see `Cuisines`
Votes

A data frame with 174 observations on the following 3 variables.

violation_code  a factor with many levels
critical_flag  is violation critical: a factor with levels NY
violation_description  violation description

A data frame with 84 observations on the following 2 variables.
cuisine_code  a character vector
cuisine_description  a character vector

Source

NYC Open Data, https://data.cityofnewyork.us/Health/DOHMH-New-York-City-Restaurant-Inspection-Results/43nn-pn8j/

See Also

ViolationCodes, Cuisines

Examples

data(Violations)
if (require(dplyr)) {
  Violations %>%
    inner_join(Cuisines, by = "cuisine_code") %>%
    filter(cuisine_description == "American") %>%
    arrange(grade_date) %>%
    head()
}

---

Votes from Scottish Parliament

<table>
<thead>
<tr>
<th>Votes</th>
<th>Votes from Scottish Parliament</th>
</tr>
</thead>
</table>

Description

Votes recorded on each ballot by each member of the Scottish Parliament in 2008 along with information about party affiliation.

Usage

Votes

Parties
Format

Voters is a data frame with 103582 rows and 3 variables.

bill  an identifier for the bill
name  the name of the member of parliament
vote  1 means a vote for, -1 a vote against. 0 is an abstention.

Parties is a data frame with 134 rows, one for each member of parliament, and 2 variables.

party  the name of the political party the member belongs to
name  the name of the member of parliament

An object of class data.frame with 134 rows and 2 columns.

Details

Almost all of the members of parliament belongs to a political party. This table identifies that party. These data were provided by Caroline Ettinger and form part of her senior honor's project at Macalester College. Prof. Andrew Beveridge supervised the thesis. Ms. Ettinger used the vote data to explore how to extract the party association of members purely from voting records. The Parties data was used to evaluate the success of methods.

world_cities  Cities and their populations

Description

A list of cities

Usage

world_cities

Format

A data frame with 4,428 observations on the following 10 variables.

game_name  integer id of record in geonames database
name  name of geographical point in plain ascii characters
latitude  latitude in decimal degrees (wgs84)
longitude  longitude in decimal degrees (wgs84)
country  ISO-3166 2-letter country code
country_region  fipscode
population  Population
timezone  the iana timezone id
modification_date  date of last modification
world_cities

Source

GeoNames: http://download.geonames.org/export/dump/

Examples

world_cities
Index

* datasets
  Cherry, 2
  CholeraDeaths, 3
  CIACountries, 4
  DataSciencePapers, 5
  Elections, 6
  Emails_train, 7
  Headlines_train, 8
  Macbeth_raw, 9
  MedicareCharges, 12
  MedicareProviders, 13
  Minneapolis2013, 14
  MLB_teams, 15
  NCI60_tiny, 16
  ordway_birds, 17
  saratoga_houses, 19
  SAT_2010, 20
  Violations, 24
  Votes, 25
  world_cities, 26
  argument (macros), 9
  babynames, 11
  Cancer, 16
  Cancer (NCI60_tiny), 16
  Cherry, 2
  CholeraDeaths, 3
  CholeraPumps (CholeraDeaths), 3
  CIACountries, 4
  CIAdata, 5
  cor, 16
  Cuisines, 24, 25
  Cuisines (Violations), 24
  DataSciencePapers, 5
  dbConnect, 22
dbConnect_scidb, 23
  dbConnect_scidb (src_scidb), 22
  Emails_test (Emails_train), 7
  Emails_train, 7
  etl_NCI60, 8
  func (macros), 9
  Headlines_test (Headlines_train), 8
  Headlines_train, 8
  index_entry, 10
  index_entry (macros), 9
  kbl, 12
  Macbeth_raw, 9
  macros, 9
  make_babynames_dist, 11
  mdsr_data (macros), 9
  mdsr_person (macros), 9
  mdsr_sql_explain_table (mdsr_table), 11
  mdsr_sql_keys_table (mdsr_table), 11
  mdsr_table, 11
  MedicareCharges, 12, 13
  MedicareProviders, 12, 13
  Minneapolis2013, 14
  MLB_teams, 15
  mysql_scidb, 23
  mysql_scidb (src_scidb), 22
  NCI60_tiny, 16
  opts_chunk, 23
  ordway_birds, 17
  Parties (Votes), 25
  pkg (macros), 9
  Rnw2Rmd, 18
  Salaries, 15
saratoga_codes (saratoga_houses), 19
saratoga_houses, 19
SAT, 20
SAT_2010, 20
save_webshot, 21
sf, 4
skim, 22
sql_func (macros), 9
sql_word (macros), 9
src_dbi, 22, 23
src_scidb, 22, 23
tbl_df, 11, 15
Teams, 15
theme_mdsr, 23
variable (macros), 9
ViolationCodes, 24, 25
ViolationCodes (Violations), 24
Violations, 24
vocab (macros), 9
Votes, 25
webshot, 21
world_cities, 26
x, 10