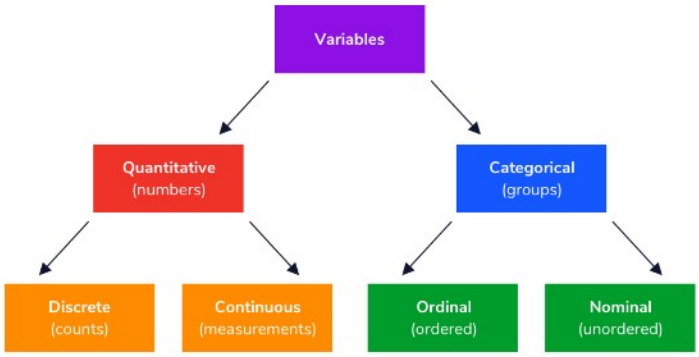
Variable Types Module

Saturday, February 5, 2022 10:25 AM

Variable Types Module

Kinds of data types for entries



See all data types in your data frame:

print(df.dtypes)

```
print(cereal.dtypes)
```

How to replace bad data entries (data cleaning)

Replace the value "missing" with a NaN entry:

Import numpy as np

df["col_name"] = df.col_name.replace("missing", np.nan)

```
auto['city-mpg'] = auto['city-
mpg'].replace(['missing'], np.nan)
print(auto['city-mpg'].unique())
```

Change a data type for a column:

df["col_name"] = df.col_name.astype("data_type")

```
auto['city-mpg'] = auto['city-
mpg'].astype('float')
```

How to create an ordering for a column (categorical data type) in your data set:

See all values:

print(movies.rating.unique())

```
print(movies['rating'].unique())
```

Create a new category with a given ordering (order only, not indexed):

movies["rating"] = pd.Categorical(movies.rating, ["G", "PG", "PG-13", "R", "UNRATED", "NOT RATED"], ordered = true)

```
movies['rating']
= pd.Categorical(movies['rating'], ['G', 'PG',
'PG-13', 'R','UNRATED', 'NOT RATED'],
ordered=True)
```

```
movies['rating']
= pd.Categorical(movies['rating'], ['G', 'PG',
'PG-13', 'R','UNRATED', 'NOT RATED'],
ordered=True)
```

Give each entry numbers (similar to enumerate() in Python):

movies["rating_codes"] = movies.rating.cat.codes

```
movies['rating_codes']
= movies['rating'].cat.codes
```

Use cat.codes to find a "median" category:

median_index = np.median(df["col_name"].cat.codes)
median_category = cat_names_list[int(median_index)]

Create a One Hot Encoding (OHE) categorical variable:

- *This allows for indexing other than the default (above) of 0,1,2,3,... This allows for a different spacing between variables, or for values that are not meant to represent an ordering. This reminds me of an opposite of .pivot because it turns the column into a binary matrix with 0's and 1's that correspond to the column type. The other columns are* **not** *dropped with this command.*

df = pd.get_dummies(data = df, columns = ["col_name"])

```
titanic = pd.get_dummies(data=titanic,
columns=['Embarked'])
print(titanic.head())
```

Education	pd.get_dummies()	Education_bachelors	Education_associates	Education_post_doc	Education_high_dip
bachelors		1	0	0	0
associates		0	1	0	0
post doctorate		0	0	1	0
post doctorate		0	0	1	0
high school diploma		0	0	0	1