UniDeep

Data Manupulate Cheat Sheet -- R vs Python

Global Setting

install.packages()	import()
getwd()	os.getcwd()
setwd()	os.chdir()
set.seed()	np.random.seed()
ls()	os.listdir()
rm()	os.remove()

Data Slicing

df[1:10,]	df.iloc[0:10,]
df[, 1:3]	df.iloc[:, 1:3]
df[, col]	df.loc[:, col]
df$col	df.col
df[df$col in c(),]	df.loc[df.col.isin([])]
df[df$col == value]	df.loc[df.col == value]
df[df$col == value, col] = value	df.loc[df.col == value,] = value
df[, -1]	df.drop(df.columns[1], 1)
df[, c(col1, col2)] = NULL	df.drop([col1, col2], 1)

Data Wrangling

df[order(col),]	df.sort_values([col])
df[order(-col),]	df.sort_values([col], ascending=[0])
colnames()	df.columns.values()
row.names()	df.index()
rbind()	pd.concat([,], axis=0)
cbind()	pd.concat([,], axis=1)
str()	df.info()
dim()	df.shape()
head()	df.head()
summary()	df.describe()
merge(, by)	pd.merge(, on)
merge(, all.x=T)	pd.merge(, how='left')
merge(, all.y=T)	pd.merge(, how='right')
merge(, all.x=T, all.y=T)	pd.merge(, how='outer')
apply(, MARGIN = 1, function(x))	df.apply(lambda x: , axis=1)
apply(, MARGIN = 2, function(x))	df.apply(lambda x: , axis=0)
tidyr::drop_na(df)	df.dropna()
df[is.na(df)] = value	df.fillna(value)
df[, mean(col3), .(col1, col2)]	df.groupby([col1, col2])[col3].mean()
df[, .N, .(col1, col2)]	df.groupby([col1, col2).size()
unique()	df.drop_duplicates()
sample_n()	df.sample()
dcast(, ~, value.var, fun)	pd.pivot_table(, index, columns, values, aggfunc)
melt(, id.vars = c(col))	pd.melt(, id_vars)

Load & Write Data

read.csv()	pd.read_csv()
write.csv(, row.name=F)	df.to_csv(, index=False)
read.table()	pd.read.table()
write.xlsx()	df.to_excel()

Data Type

as.numeric()	as.numeric()
as.character()	as.character()
as.factor()	as.factor()
as.data.frame()	as.data.frame()
as.Date()	as.Date()

Basic Function

seq()	range()
rep()	np.repeat()
length()	len()
table()	pd.crosstab()
unique()	set()
class()	type()
strsplit()	str.split()
is.null()	pd.isnull()
is.na()	pd.isnan()
paste(A, B, sep='_')	%s_%s' %(A, B)
setdiff(A, B)	[x for x in A if x not in B]
grep()	re.findall()

Basic Statistics

mean()	np.mean()
rowSums()	df.sum(1)
colSums()	df.sum(0)
colMeans()	df.mean(0)
rowMeans()	df.mean(1)
var()	np.var()
sd()	np.std()
t.test()	sp.stats.ttest_ind()
rnorm()	sp.stats.norm.rvs()
dnorm()	sp.stats.norm.pdf()
lm()	linear_model.LinearRegression()