

Python H2OFrame / Pandas DataFrame Munging Conversion Table

Note: A blank under the **Pandas Equivalent Method** means the method is equivalent to H2O. (Parenthesis are not always shown when need).

Last updated on 7/20/2017. If you notice an missing method, please submit a pull request with the addition or post a message to the h2ostream Google Group.

H2OFrame Method	Pandas Equivalent Method
.abs	
.acos	.apply(lambda x: numpy.arccos(x), axis = 0)
.acosh	.apply(lambda x: numpy.arccosh(x), axis = 0)
.all	
.any	
.any_na_rm	
.anyfactor	
.apply	.apply
.as_data_frame	
.as_date	.to_datetime
.ascharacter	astype(str)
.asfactor	.astype('category') or .astype('object')
.asin	.apply(lambda x: numpy.arcsin(x), axis = 0)
.asinh	.apply(lambda x: numpy.arcsinh(x), axis = 0)
.asnumeric	astype(numpy.float) or apply(numpy.float)
.atan	.apply(lambda x: numpy.arctan(x), axis = 0)
.atanh	.apply(lambda x: numpy.arctanh(x), axis = 0)
.categories	.unique()
.cbind	.concat()
.ceil	.apply(numpy.ceil)
.col_names	.columns
.columns	
.columns_by_type	.select_dtypes()
.concat	
.cor	.corr
.cos	.apply(lambda x: numpy.arccos(x), axis = 0)
.cosh	.apply(lambda x: numpy.arccos(x), axis = 0)
.cospi	.apply(lambda x: numpy.cos(numpy.pi * x), axis = 0)
.count	
.countmatches	.str.contains()
.cummax	
.cummin	
.cumprod	
.cumsum	
.cut	
.day	Series.dt.day
.dayOfWeek	DatetimeIndex(pandas_dataframe[time_column]).dayofweek
.ddply	
.describe	
.difflag1	.diff
.digamma	scipy.special.digamma()
.dim	.shape
.drop	
.entropy	NA
.exp	numpy.exp()
.expm1	numpy.expm1()

H2OFrame Method	Pandas Equivalent Method
.filter_na_cols	NA
.flatten	
.floor	.apply(numpy.floor)
.frame	NA
.frame_id	NA
.from_python	NA
.gamma	scipy.special.gamma()
.get_frame	NA
.get_frame_data	similar to the purpose of to_csv()
.getrow	list(pandas_dataframe.loc[0,:])
.group_by	.groupby()
.gsub	.replace()
.head	
.hist	
.hour	DatetimeIndex(pandas_dataframe[time_column]).year
.ifelse	numpy.where()
.impute	NA
.insert_missing_values	NA
.interaction	NA
.isax	NA
.ischaracter	.instance(pandas_column, object)
.isfactor	NA
.isin	
.isna	.isnull
.isnumeric	NA
.isstring	.instance(pandas_column, object)
.kfold_column	NA
.kurtosis	
.levels	.cat.categories, .unique()
.lgamma	scipy.special.gammaln()
.log	numpy.log()
.log10	numpy.log10()
.log1p	numpy.log1p()
.log2	numpy.log2()
.logical_negation	numpy.logical_not()
.lstrip	.str.lstrip("")
.match	
.max	
.mean	
.median	
.merge	
.min	
.mktime	
.mode	NA
.modulo_kfold_column	NA
.moment	pd.to_datetime()
.month	Series.dt.month
.mult	.dot
.na_omit	.dropna()
.nacct	.isnull().sum()
.names	.columns
.nchar	.str.len()

H2OFrame Method	Pandas Equivalent Method
.ncol	.shape[1]
.ncols	.shape[1]
.nlevels	.nunique()
.nrow	.shape[0]
.nrows	.shape[0]
.num_valid_substrings	
.pop	
.prod	
.quantile	
.rbind	
.refresh	
.relevel	NA
.rep_len	NA
.round	
.rstrip	.str.rstrip()
.runif	numpy.random.uniform()
.scale	sklearn.preprocessing.StandardScaler()
.sd	.std
.set_level	NA
.set_levels	NA
.set_name	.rename()
.set_names	.rename()
.shape	
.show	NA
.sign	numpy.sign()
.signif	NA
.sin	.apply(lambda x: numpy.sin(x), axis = 0)
.sinh	.apply(lambda x: numpy.sinh(x), axis = 0)
.sinpi	.apply(lambda x: numpy.sin(numpy.pi * x, axis = 0)
.skewness	.skew
.split_frame	NA
.sqrt	.apply(lambda x: numpy.sqrt(x), axis = 0)
.ss	NA
.stratified_kfold_column	sklearn.model_selection.StratifiedKFold
.stratified_split	sklearn.model_selection.StratifiedShuffleSplit
.strsplit	.str.split
.structure	NA
.sub	.str.replace()
.substring	.str.slice()
.sum	
.summary	.describe()
.table	.value_counts()
.tail	
.tan	.apply(lambda x: numpy.tan(x), axis = 0)
.tanh	.apply(lambda x: numpy.tanh(x), axis = 0)
.tanpi	.apply(lambda x: numpy.tan(numpy.pi * x, axis = 0)
.tolower	
.toupper	.apply(lambda x: x.upper(), inplace=True)
.transpose	
.trigamma	scipy.special.polygamma(x,3)
.trim	.str.strip
.trunc	

H2OFrame Method	Pandas Equivalent Method
.type	.dtype
.types	.dtypes
.unique	
.var	
.week	Series.dt.week
.which	NA
.year	Series.dt.year