Cheat Sheet: Exploratory Data Analysis

IBM Developer
SKILLS NETWORK

Command	Syntax	Description	Example
summarize()	`summarize(.data, ...)`	`summarize` function reduces a data frame to a summary of just one vector or value. .data A data frame, data frame extension (e.g. a tibble), or a lazy data frame ... Name-value pairs of summary functions. The name will be the name of the variable in the result. The value should be an expression that returns a single value like min(x), n(), or sum(is.na(y))	`avg_delays <- sub_airline %>% group_by(Reporting_Airline, DayOfWeek) %>% summarize(mean_delays = mean(ArrDelayMinutes), .groups = 'keep')`
group_by()	`group_by(.data, ..., .add = FALSE, .drop = group_by_drop_default(.data))`	`group_by` function takes an existing table and converts it into a grouped table where operations are performed "by group". .data A data frame, data frame extension (e.g. a tibble), or a lazy data frame .add When FALSE, the default, group_by() will override existing groups. .drop Drop groups formed by factor levels that don't appear in the data	`sub_airline %>% group_by(Reporting_Airline) %>% summarize(mean_delays = mean(ArrDelayMinutes))`
cor()	`cor(x, use=, method= )`	`cor` function computes the correlation coefficient x: Matrix or data frame use: Specifies the handling of missing data. method: Specifies the type of correlation. Options are pearson, spearman or kendall.	`sub_airline %>% select(DepDelayMinutes, ArrDelayMinutes) %>% cor(method = "pearson")`

cor.test()	`cor.test(x, y, alternative = c("two.sided", "less", "greater"), method = c("pearson", "kendall", "spearman"), exact = NULL, conf.level = 0.95, continuity = FALSE, …)`	`cor.test` function is a test for association/correlation between paired samples. It returns both the correlation coefficient and the significance level(or p-value) of the correlation . x, y: numeric vectors of data values. x and y must have the same length.	`sub_airline %>% cor.test(~DepDelayMinutes + ArrDelayMinutes, data = .)`

aov	`aov(formula, data = NULL, projections = FALSE, qr = TRUE, contrasts = NULL, …)`	`aov` function (Analysis of Variance (ANOVA)) is a statistical method used to test whether there are significant differences between the means of two or more groups.
		formula: A formula specifying the model.
		data: A data frame in which the variables specified in the formula will be found. If missing, the variables are searched for in the standard way.
			`aa_as_subset <- sub_airline %>% select(ArrDelay, Reporting_Airline) %>% filter(Reporting_Airline == 'AA' | Reporting_Airline == 'AS')`
			`ad_aov <- aov(ArrDelay ~ Reporting_Airline, data = aa_as_subset)`

count()	`count(df, vars = NULL, wt_var = NULL)`	`count` function lets you quickly count the unique values of one or more variables df: data frame to be processed vars: variables to count unique values of	`sub_airline %>% count(Reporting_Airline)`
ggplot()	`ggplot(data = NULL, mapping = aes(), ..., environment = parent.frame())`	`ggplot` function initializes a ggplot object. It can be used to declare the input data frame for a graphic and to specify the set of plot aesthetics intended to be common throughout all subsequent layers unless specifically overridden.	`ggplot(aes(x = Reporting_Airline, y = DayOfWeek, fill = mean_delays))`
corrplot()	`corrplot(method=, type=,....)`	`corrplot` function provides a visual exploratory tool on correlation matrix that supports automatic variable reordering to help detect hidden patterns among variables. method: There are seven visualization methods (parameter method) in corrplot package, named 'circle', 'square', 'ellipse', 'number', 'shade', 'color', 'pie' type: There are three layout types (parameter type): 'full', 'upper' and 'lower'.	`corrplot(airlines_cor, method = "color", col = col(200), type = "upper", order = "hclust", addCoef.col = "black", # Add coefficient of correlation tl.col = "black", tl.srt = 45, #Text label color and rotation )`
geom_bar()	`geom_bar(mapping = NULL, data = NULL, stat = "bin", position = "stack", ...)`	`geom_bar` function is used to produce 1d area plots: bar charts for categorical x, and histograms for continuous y.	`ggplot(aes(x = Reporting_Airline, y = Average_Delays)) + geom_bar(stat = "identity") + ggtitle("Average Arrival Delays by Airline")`

geom_tile()	`geom_tile(mapping = NULL, data = NULL, stat = "identity", position = "identity", ...)`	`geom_tile` function tile plane with rectangles.	`ggplot(avg_delays, aes(x = Reporting_Airline, y = lubridate::wday(DayOfWeek, label = TRUE), fill = bins)) + geom_tile(colour = "white", size = 0.2)`
geom_text()	`geom_text(mapping = NULL, data = NULL, stat = "identity", position = "identity", parse = FALSE, ...)`	`geom_text` used for text annotation.	`ggplot(avg_delays, aes(x = Reporting_Airline, y = lubridate::wday(DayOfWeek, label = TRUE), fill = bins)) + geom_tile(colour = "white", size = 0.2) + geom_text(aes(label = round(mean_delays, 3)))`

labs()	`labs(...)`
	`... a list of new names in the form aesthetic = "new name"`
		`labs` Change axis labels and legend titles	`ggplot(avg_delays, aes(x = Reporting_Airline, y = lubridate::wday(DayOfWeek, label = TRUE), labs(x = "Reporting Airline",y = "Day of Week",title = "Average Arrival Delays") fill = bins)) +`

scale_fill_manual()	`scale_fill_manual(..., values)`	`scale_fill_manual` function Change axis labels and legend titles ... common discrete scale parameters: name, breaks, labels, na.value, limits and guide. See discrete_scale for more details values: a set of aesthetic values to map data values to.	`scale_fill_manual(values = c("#d53e4f", "#f46d43", "#fdae61", "#fee08b", "#e6f598", "#abdda4"))`

Author(s)

Lakshmi Holla

Changelog

Date	Version	Changed by	Change Description
2021-08-09	1.0	Lakshmi Holla	Initial Version