**1. Data Cleaning and Transformation**

**Functions for String Manipulation**

1. **stringr package**:
   * **str\_detect()**: Detect patterns in strings.
     + **Use case**: Filter rows with specific text patterns.
     + **Example**:

R

Copy code

library(stringr)

contains\_keyword <- str\_detect(data$comments, "important")

* + - **Study**: Explore regular expressions (regex) to fully leverage this function.

1. **str\_split()**: Split strings into components.
   * **Use case**: Parse data (e.g., splitting "John\_Doe" into "John" and "Doe").
   * **Example**:

R

Copy code

split\_names <- str\_split("John\_Doe", "\_")

**Functions for Missing Data**

1. **mice package**:
   * **mice()**: Perform multiple imputations.
     + **Use case**: Handle missing values in numerical or categorical data.
     + **Example**:

R

Copy code

library(mice)

imputed\_data <- mice(data, method = "pmm")

* + - **Study**: Understand imputation models (predictive mean matching, logistic regression).

1. **naniar package**:
   * **vis\_miss()**: Visualize missing data patterns.
     + **Use case**: Understand where and how much data is missing.
     + **Example**:

R

Copy code

library(naniar)

vis\_miss(data)

**2. Statistical Analysis**

**Functions for Advanced Statistical Models**

1. **glmnet package**:
   * **cv.glmnet()**: Cross-validated LASSO and Ridge regression.
     + **Use case**: Handle multicollinearity in numerical predictors.
     + **Example**:

R

Copy code

library(glmnet)

fit <- cv.glmnet(as.matrix(data[,-1]), data$target, alpha = 1)

* + - **Study**: Learn about L1/L2 regularization techniques.

1. **lme4 package**:
   * **lmer()**: Fit linear mixed-effects models.
     + **Use case**: Analyze hierarchical or grouped data.
     + **Example**:

R

Copy code

library(lme4)

fit <- lmer(score ~ time + (1|group), data = data)

* + - **Study**: Understand random vs. fixed effects.