**Bioinformatics Programming: Glossary and Exam Guide**

**Key Terms and Concepts**

**Module 1: R Basics**

* **Vector**: One-dimensional array holding elements of the same type
* **List**: Ordered collection that can contain elements of different types
* **Matrix**: Two-dimensional array holding elements of the same type
* **Data Frame**: Two-dimensional structure that can hold different types of data in different columns
* **Factor**: Special data type for categorical variables
* **Working Directory**: Current location in the file system where R reads and writes files

**Module 2: Advanced R Programming**

* **Logical Vector**: Vector containing TRUE/FALSE values
* **Correlation**: Measure of the strength and direction of relationship between variables
* **Linear Model**: Statistical model describing relationship between variables using linear equation
* **Residuals**: Differences between observed and predicted values in a model
* **P-value**: Probability value indicating statistical significance
* **Type I Error**: False positive (rejecting true null hypothesis)
* **Type II Error**: False negative (failing to reject false null hypothesis)

**Module 3: Python for Bioinformatics**

* **NumPy**: Library for numerical computing in Python
* **Pandas**: Library for data manipulation and analysis
* **DataFrame**: 2-dimensional labeled data structure
* **Series**: 1-dimensional labeled array
* **Array**: N-dimensional homogeneous data structure

**Module 4: Bioinformatics Applications**

* **RNA-seq**: High-throughput sequencing of RNA
* **Bulk RNA-seq**: RNA sequencing of tissue samples
* **Single-cell RNA-seq**: RNA sequencing of individual cells
* **Normalization**: Process of adjusting data to make samples comparable
* **Differential Expression**: Analysis to find genes with different expression levels between conditions
* **UMAP**: Uniform Manifold Approximation and Projection for dimension reduction
* **PCA**: Principal Component Analysis

**Potential Exam Questions and Answers**

1. **Q**: How do you check if a correlation is statistically significant? **A**: Use cor.test() function in R, which provides both correlation coefficient and p-value:

r

Copy

result <- cor.test(x, y)

*# Check p-value*

if(result$p.value < 0.05) {

print("Significant correlation")

}

1. **Q**: What is the difference between a matrix and a data frame in R? **A**: A matrix must contain elements of the same type (all numeric or all character), while a data frame can contain different types in different columns. Matrix is typically used for mathematical operations, while data frame is for heterogeneous data analysis.
2. **Q**: How do you identify outliers in a dataset? **A**: Common method is using 3 standard deviations from mean:

r

Copy

outliers <- abs(x - mean(x)) > 3\*sd(x)

1. **Q**: How do you evaluate the goodness of fit in a linear model? **A**: Check:
   * R-squared value from summary(model)
   * Residual plots
   * Q-Q plot for normality of residuals
   * F-statistic and its p-value
2. **Q**: How do you filter genes in RNA-seq analysis? **A**: Common criteria include:
   * Minimum count threshold
   * Minimum number of samples expressing the gene
   * Remove genes with zero variance

r

Copy

*# Example in DESeq2*

keep <- rowSums(counts(dds)) >= 10

dds <- dds[keep,]

1. **Q**: What are the assumptions of linear regression? **A**: Key assumptions include:
   * Linearity of relationship
   * Independence of errors
   * Homoscedasticity (constant variance)
   * Normality of residuals
   * No perfect multicollinearity
2. **Q**: How do you handle missing values in R? **A**: Common approaches:

r

Copy

*# Remove NA values*

clean\_data <- na.omit(data)

*# Check for NA*

is.na(data)

*# Replace NA with mean*

data[is.na(data)] <- mean(data, na.rm=TRUE)

1. **Q**: How do you perform quality control in single-cell RNA-seq? **A**: Key steps include:
   * Filter cells by number of genes expressed
   * Filter genes by number of cells expressing them
   * Check percentage of mitochondrial genes
   * Remove doublets

python

Copy

sc.pp.filter\_cells(adata, min\_genes=200)

sc.pp.filter\_genes(adata, min\_cells=3)

**Exam Tips**

1. **Code Implementation**
   * Always check your data types
   * Use proper error handling
   * Comment your code
   * Test edge cases
2. **Statistical Analysis**
   * Check assumptions before applying tests
   * Always interpret p-values in context
   * Consider multiple testing correction when necessary
3. **Data Visualization**
   * Choose appropriate plot types
   * Label axes and titles clearly
   * Consider color-blind friendly palettes
4. **Common Pitfalls**
   * Not checking data quality before analysis
   * Forgetting to handle missing values
   * Not considering assumptions of statistical tests
   * Ignoring outliers
5. **Best Practices**
   * Document your analysis steps
   * Use version control
   * Keep track of data transformations
   * Validate results