Document Summary

This document provides a comprehensive introduction to regression analysis within the context of supervised machine learning. It begins by defining supervised learning, differentiating between classification (predicting discrete labels) and regression (predicting continuous values), and outlining the key components: input features (X), output labels (Y), training data, and the learning algorithm. The core focus then shifts to regression analysis, a technique for predicting continuous outcomes. Several regression types are introduced: linear regression (modeling a linear relationship between dependent and independent variables using a straight line, Y = bX + c), multiple regression (extending linear regression to multiple input variables), polynomial regression (fitting a curve to data), and logistic regression (used for classification despite its name, predicting probabilities). A simple linear regression example is provided, illustrating the relationship between a marketing company's advertisement spending and resulting sales. The document then contrasts regression algorithms in machine learning (focused on prediction and generalization) with regression using statistical models (focused on understanding relationships). It details how to measure linear relationships using Pearson's correlation coefficient and perform linear regression using the `statsmodels` library in Python, emphasizing the interpretation of the R-squared value, p-values, and coefficients in the statistical summary. The meaning and significance of coefficients are further explained for linear, logistic, and non-linear models, highlighting their magnitude, sign, and statistical significance. A practical application of simple linear regression is demonstrated using a sample dataset predicting house prices based on house size. The methodology (hypothesis setup, data collection, model fitting, evaluation, and prediction) is clearly outlined, along with the Python code implementing the model using `sklearn`. This section visually represents the regression line fitted to the data, and the prediction of new house prices. The document further explores polynomial regression, extending linear regression to model non-linear relationships using polynomial functions. A step-by-step guide, including Python code, demonstrates how to perform polynomial regression, transform features, fit the model, make predictions, and visualize the results. This section emphasizes the use of `PolynomialFeatures` from `sklearn` to transform input features into polynomial features before applying linear regression. Finally, the concepts of generalization, overfitting, and underfitting are discussed. Generalization, the ability of a model to perform well on unseen data, is highlighted as a crucial goal. Overfitting (high accuracy on training data, low accuracy on unseen data) and underfitting (low accuracy on both) are defined, along with their causes and solutions (regularization, pruning, cross-validation, early stopping, ensembling for overfitting; increasing model complexity, feature engineering, increasing training time, hyperparameter tuning, and improving data quality for underfitting). The document concludes with a detailed example of building and evaluating a linear regression model in Python using the Boston Housing dataset, covering data preparation, train-test splitting, model evaluation (intercept, slope, MSE,

MAE), visualization, prediction, and comparison of actual and predicted values.