Project Name: Supply Chain - UPH
Project Type: Personal/Assigned project

Project Goals:
“We manage what we measure, but frequently we measure what is easy" inspired by this quote I found a Kaggle dataset with units and pick time to evaluate. Exploring the data I found that there was minimal variance in many features I expected to be predictive. I found using a polynomial features model with order complexity was 65% more accurate in predicting the time needed to pick an order.

Deliverables: Well documented Jupyter Notebook, slide deck, and 5 min presentation

Project Inspiration:
“We manage what we measure, but frequently we measure what is easy.”

In my experience this is too frequently true and can be disastrous when done with metrics that are directly used to reward or penalize employees as is commonly done with UPH (units per hour) metrics for measuring employee productivity. This is an easy metric to calculate, how many units (boxes) did an employee move/get/create divided by the number of hours worked. However, it fails to account for many other factors that create variation in productivity. The type of items being picked, the size and weight, the location (transit time) involved, location (indoor/outdoor), the method of movement (motorized vs. walking), size of the area the items are located within, etc. are examples of physical variations that are independent of personal motivation to perform well. Even if these are controlled for, the tenure of the operator (amount of experience), number of hours worked per shift, etc. are additional factors that may impact performance that are also independent of motivation.

And, because this measure is easily obtained and communicated it is also frequently the only measure of performance that employees are made aware of with daily frequency and as a result becomes defined as the most important measure of successful performance. Frequently to the exclusion of accuracy or safety.

Could a model be built to create a more realistic productivity baseline where these additional factors are taken into account? Do these additional factors increase the accuracy of productivity prediction?

Project Scope:
Based on publicly available data what is the baseline productivity predicted?
How accurate is the baseline?
Does adding additional features of tenure and shift length improve accuracy of model predictions?

Stage	Tools	Brief Description of Process	Challenge Resolution
Plan	• Kaggle	• Found a Kaggle dataset with suggested questions of: What days are busiest? Are things getting better or worse? Who is the best? Who's not pulling their weight? • This inspired additional questions: What are the underlying environmental factors that are impacting these workers? • Initial scope wanted to look at factors other than operator pace that may impact productivity: environmental factors, equipment, layout, worker training and/or tenure, product mix, seasonality, etc. • Researched multiple public databases for warehouse productivity data • Narrowed scope based on data available to examine order complexity and operator tenure	• Searched Data World, GitHub, and Kaggle for a dataset with measures that could be used for productivity
Acquire	• Visual Studio • .py script	• Dataset shape (159980, 5) • No data definitions, had to determine/define column variables	• Limited information available: pick start time, pick end time, # boxes, # lines, operator • Decided to create

			additional features during prepare to explore the data
Prepare	- Visual Studio .py script - Jupyter Notebook - Tableau	- Extracted additional fields based on start and end time fields: Hour, Day, Week, Month, Year, etc - Removed operators who appeared < 9x in the dataset - Reduced year range to only full years - Removed erroneous data like negative time or boxes or lines - Created calculated fields: pick seconds, seconds/line, seconds/box - Split data into train, validate, test datasets for future modeling	Had to learn more about timestamp syntax and how to extract components
Explore	- Jupyter Notebook - Visual Studio	- Found orders with more than 1 box accounted for only 2K of 96K records, removed these from data for modeling to reduce noise - Visualized pick seconds (time to fulfill order) and pick lines (order complexity) by hour/day/week/month/year looking for patterns or seasonality trends - Further created datasets by operator and visualized trends by operator shift length (FT/PT) and operator tenure (over how long a period of time does that operator appear in the dataset	
Model	- Jupyter Notebook - Visual	Established baseline using mean pick seconds and calculated the MAE (median absolute error) using that prediction	

	Studio	• 1st round of modeling used linear regression and polynomial regression with features of was the order picked in the last hour of the day (bool) and number of lines per order (a measure of order complexity) • 2nd round of modeling used operator tenure and shift length features • 3rd round of modeling: used operator tenure, shift length, and order complexity as model features	
Evaluate	• Jupyter Notebook • Visual Studio	• For the 1st round: Compared accuracy of predictions with features of was the order picked in the last hour of the day (bool) and number of lines per order (a measure of order complexity) vs. using only lines per order. No significant change in prediction accuracy when using only lines per order vs. both features • Found using lines per order (order complexity) was 65% better at predicting the seconds per order than using mean • 2nd Round: using operator tenure and/or shift length produce prediction accuracy equivalent to using mean pick time • No improvement over baseline • 3rd round: No improvement adding operator features with order complexity • Determined order complexity was driving feature for predicting order pick time required	•
Model	**How does your**	• Polynomial regression worked best	•

Explanation	**algorithm work?**	- Polynomial regression fits a nonlinear relationship between the value of x and the corresponding conditional mean of y - Although polynomial regression fits a nonlinear model to the data, as a statistical estimation problem it is linear. For this reason, polynomial regression is considered to be a special case of multiple linear regression	
Delivery	- Jupyter Notebook - Canva	●	●