

Homework #2

Submitted by: Aditi Shashank Joshi (1222838916)

What is the Design of Experiments?

Design of experiments (DOE) is a systematic, efficient method that enables scientists and engineers to study the relationship between multiple input variables (aka factors) and key output variables (aka responses). It is a structured approach for collecting data and making discoveries.

[1] In this assignment, I will use pairwise combination testing.

Software Specification

I will be working on Dominos pizza delivery software. The specifications listed below only estimate the time required for a successful delivery and do not address other aspects of the application.

The following are the factors I considered when estimating the time for pizza delivery by delivery agent are:

	<u>Factors</u>				
<u>Different Values</u>	Vehicle for commute	Delivery Agent Rating	Weather	Distance	Traffic
	Bicycle	1	Sunny	Near	No Traffic
	Bike	2	Light Rain	Average	Low Traffic
	Car	3	Heavy Rain	Far	Regular Traffic
		4	Humid		High Traffic
		5	Fog		

1. Vehicle for Commute:

This denotes which vehicle was being used by the delivery agent to deliver the food to the customers. This is independent of the delivery agent, and not related to the application. For now we have considered three vehicles.

Input Domain - [Bicycle, Bike, Car]

2. Delivery Agent Rating:

This is a measure of how well the delivery agent has previously performed. Consumers rate delivery agents who deliver orders on time or ahead of schedule higher than those who deliver orders late. Based on previous performance, the delivery agent is likely to perform accordingly. The lowest level of rating is 1, and the highest level of rating is 5.

Input Domain - [1,5]

3. Weather:

This is a measurement of the weather conditions at the time of delivery. Some circumstances are known to favor faster delivery than others. If the weather is bad, the delivery will most likely take longer than usual.

Input Domain - [Sunny, Light Rain, Heavy Rain, Humid, Fog]

4. Distance:

This is a categorical classification of the distance between the restaurant where the order was placed and the destination (delivery address).

Input Domain - [Near, Average, Far]

5. Traffic:

This is a measure of the amount of traffic on the way to the destination. This value is likely to be higher than usual during peak hours (morning/evening) and will affect delivery time.

Input Domain - [No Traffic, Low Traffic, Regular Traffic, High Traffic]

Tool Reports

I am using Pairwiser by Inductive which is a free web-based software for generating pairwise combination tests. Pairwiser has an easy to use web UI that allows you to define the parameters and input of your system under test. Whether you are testing a Web UI, a product line or a highly configurable system, you can define your parameters and inputs and constraints between them and generate tests. [4]

Why do we need a Pairwiser tool? [5]

1. Web-based easy to use.
2. Generate the combination and export to excel.
3. Generate the Test case and export to JIRA directly.
4. You can apply the constraint to the input parameter.

We rearrange the parameters in order to generate maximum test cases, and then we put the test cases in the “Define Parameters” section of the Pairwiser tool.

Pairwiser by Inductive

Test Plan: Delivery system [new] [copy] [edit] [delete] [examples]

Intro [x] Define Parameters [x] Required Tests [x] Generate Tests [x] Analysis of Tests [x] Test Script Template [x] Test Scripts [x]

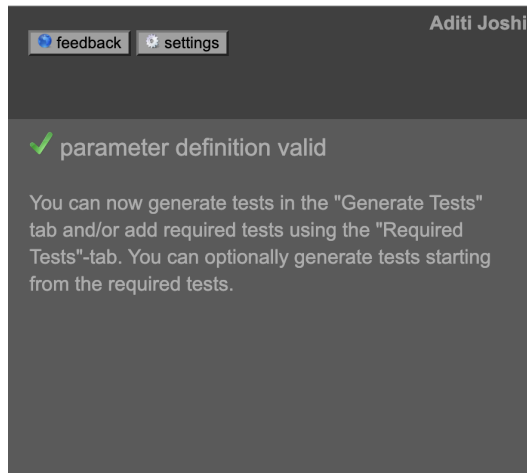
Define Parameters [save] [parameter] [constraint] [Import from CSV]

Parameters:

1.	▲▼	Delivery Agent Rating	[+]	[x]	1	[x]	2	[x]	3	[x]	4	[x]	5	[+]
2.	▲▼	Weather	[+]	[x]	Sunny	[x]	Light Rain	[x]	Heavy Rain	[x]	Humid	[x]	Fog	[+]
3.	▲▼	Vehicle for Commute	[+]	[x]	Bicycle	[x]	Bike	[x]	Car	[x]	[+]			
4.	▲▼	Distance	[+]	[x]	Near	[x]	Average	[x]	Far	[x]	[+]			
5.	▲▼	Traffic	[+]	[x]	No Traffic	[x]	Low Traffic	[x]	Regular Traffic	[x]	High Traffic	[x]	[+]	

[+ parameter]

We can even see if the parameters are valid by checking them on Pairwise



The total number of combinations generated from the table is 900 ($5 \times 5 \times 3 \times 3 \times 4$), but utilizing the pairwise tool, this number is substantially reduced from 900 to 32. The DOE tool known as Pairwise tool generated these 32 combinations. The pairwise combination testing satisfies all of the test cases created by the DOE tool.

A	B	C	D	E	F	G
Tests \ Para	Delivery Age	Weather	Vehicle for C	Distance	Traffic	
1	4	Light Rain	Bicycle	Average	High Traffic	
2	2	Light Rain	Bicycle	Average	No Traffic	
3	4	Light Rain	Bike	Far	No Traffic	
4	1	Humid	Car	Far	Low Traffic	
5	3	Sunny	Bicycle	Near	Low Traffic	
6	1	Heavy Rain	Bike	Average	No Traffic	
7	1	Fog	Bike	Average	High Traffic	
8	4	Humid	Bicycle	Near	Regular Traffic	
9	1	Fog	Bicycle	Near	Low Traffic	
10	5	Light Rain	Car	Far	Regular Traffic	
11	2	Sunny	Bicycle	Far	Low Traffic	
12	3	Fog	Car	Average	No Traffic	
13	4	Heavy Rain	Car	Near	Regular Traffic	
14	5	Sunny	Bike	Average	Regular Traffic	
15	3	Humid	Bike	Average	No Traffic	
16	5	Heavy Rain	Bicycle	Near	High Traffic	
17	3	Light Rain	Car	Far	High Traffic	
18	5	Light Rain	Bike	Near	No Traffic	
19	2	Sunny	Car	Far	No Traffic	
20	5	Fog	Bike	Near	Low Traffic	
21	3	Fog	Bicycle	Far	Regular Traffic	
22	1	Heavy Rain	Bike	Far	Regular Traffic	
23	2	Sunny	Bike	Near	High Traffic	
24	2	Heavy Rain	Bike	Average	Low Traffic	
25	2	Fog	Car	Far	Regular Traffic	
26	4	Fog	Bike	Average	Low Traffic	
27	5	Humid	Bike	Far	High Traffic	
28	4	Sunny	Bicycle	Average	High Traffic	
29	1	Light Rain	Bicycle	Far	Low Traffic	
30	3	Heavy Rain	Car	Far	High Traffic	
31	2	Humid	Bicycle	Far	Regular Traffic	
32	1	Sunny	Bike	Far	Low Traffic	

By checking the coverage, we can see the analysis of test cases and the coverage. For 2-wise analysis, if this number is 100%, it means that all possible and valid pairs of value assignments are in at least one of the tests. If this number is lower, it means that some valid pairs of values are not present in the tests.

2-wise Coverage:

Tests: 32

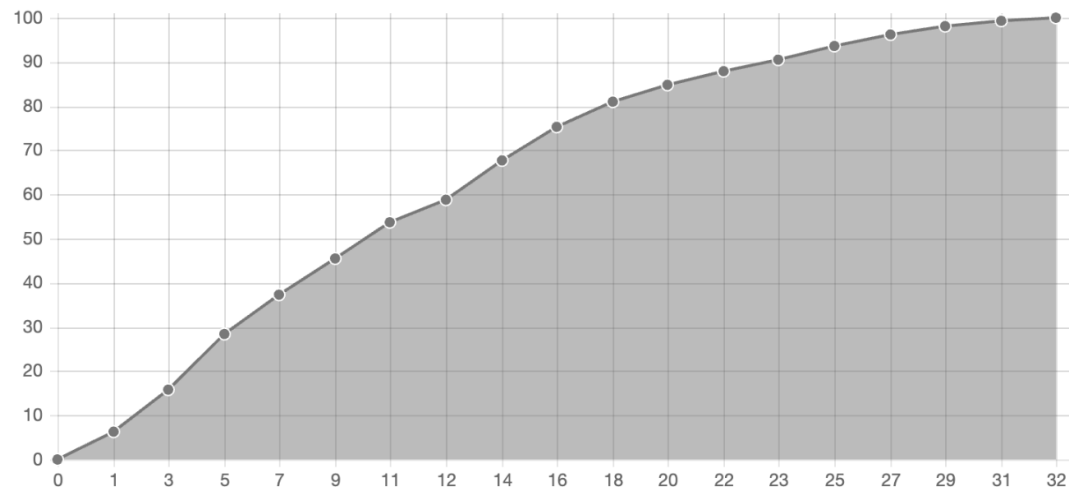
Number of Possible Tests ⓘ: 900

Coverage ⓘ: 100%

Valid combinations of 2 values ⓘ:

- covered ⓘ: 158
- total ⓘ: 158
- uncovered ⓘ: 0

Accumulated Coverage ⓘ:



Test cases

There are over 900 combinations possible when all factors and levels are taken into account. This number is bound to rapidly increase with any number of factors or levels. In order to minimize this count, I will execute all test cases that are generated based on pairwise combinations. These test cases seem to cover all possible pairs of values between two factors.

	Delivery Agent Rating	Weather	Vehicle for Commute	Distance	Traffic
1	4	Light Rain	Bicycle	Average	High Traffic
2	2	Light Rain	Bicycle	Average	No Traffic
3	4	Light Rain	Bike	Far	No Traffic
4	1	Humid	Car	Far	Low Traffic
5	3	Sunny	Bicycle	Near	Low Traffic
6	1	Heavy Rain	Bike	Average	No Traffic
7	1	Fog	Bike	Average	High Traffic
8	4	Humid	Bicycle	Near	Regular Traffic
9	1	Fog	Bicycle	Near	Low Traffic
10	5	Light Rain	Car	Far	Regular Traffic
11	2	Sunny	Bicycle	Far	Low Traffic
12	3	Fog	Car	Average	No Traffic
13	4	Heavy Rain	Car	Near	Regular Traffic
14	5	Sunny	Bike	Average	Regular Traffic
15	3	Humid	Bike	Average	No Traffic
16	5	Heavy Rain	Bicycle	Near	High Traffic
17	3	Light Rain	Car	Far	High Traffic
18	5	Light Rain	Bike	Near	No Traffic
19	2	Sunny	Car	Far	No Traffic
20	5	Fog	Bike	Near	Low Traffic
21	3	Fog	Bicycle	Far	Regular Traffic
22	1	Heavy Rain	Bike	Far	Regular Traffic
23	2	Sunny	Bike	Near	High Traffic
24	2	Heavy Rain	Bike	Average	Low Traffic
25	2	Fog	Car	Far	Regular Traffic
26	4	Fog	Bike	Average	Low Traffic
27	5	Humid	Bike	Far	High Traffic
28	4	Sunny	Bicycle	Average	High Traffic
29	1	Light Rain	Bicycle	Far	Low Traffic
30	3	Heavy Rain	Car	Far	High Traffic
31	2	Humid	Bicycle	Far	Regular Traffic
32	1	Sunny	Bike	Far	Low Traffic

References

[1] Design of experiments

https://www.jmp.com/en_ph/statistics-knowledge-portal/what-is-design-of-experiments.html

[2] Real life examples of how to use pairwise technique in test design

<https://blog.issart.com/real-life-examples-of-how-to-use-pairwise-technique-in-test-design/>

[3] Design of Experiments (DOE)

<https://www.moresteam.com/toolbox/design-of-experiments.cfm>

[4] Pairwiser: Pairwise Testing and Test Generation Tool <https://inductive.no/pairwiser/>

[5] Testing Techniques -Pairwise Testing using Pairwiser Tool

<https://scrolltest.com/2017/01/14/testing-techniques-pairwise-testing/>