Machine Learning

- **Day 1: Prerequisites: What is ML?**
- **Day.iloc[2:-1] : Azure ML**
- **Day[-1] : Big Data ML -> Databricks**

- **Process**

Prerequisite: Big Data

Data at rest:SQL, NoSQL, Files

Data in motion

2 kind of datastore:

1. **Transactional- OLTP (Online Tx Process)**
1. **SQL Server, mySQL, Maria, PostgreSQL, MongoDB, GraphQL**
1. **SQL or NoSQL**
2. **CRUD ops-> editing data!**
3. **Poor performance = SEARCHING**
2. **Analytical- OLAP (Online Analytical Process)**
1. **SQL DW, Hive, HBase, Cassandra**
1. **SQL or NoSQL**
2. **SEARCH-> inserting and searching data!**
3. **Poor performance = Editing**

Big data = too big for 1 machine!
1 machine = 2 numbers!

1, 2, 3

M1-> 1,2 — replicate— M3-> 1,2
M2-> 3
M4-> 3

Edit-> 2 into 4
Assume: every ops-> 1ms

	1ms	2	3	4
M1	**NF**	**F**	**R**	
M2	**NF**	**-**		

Total cost= 5ms, longest= 3ms

$(Infrastructure) > $(data)!

Sharding

M1-> 1,2 M2-> 2,3 M3-> 3, 1

(Edit-> 2 to 4)

	1ms	2	3	4
M1	**NF**	**F**	**R**	**-**
M2	**F**	**R**	**NF**	**-**
M3	**NF**	**NF**	**-**	

Total cost= 8ms, longest= 3ms

Search for 2:
M1: 1,2. M2: 3

OLTP:

	1ms	2	3	4
M1	**NF**	**F**		
M2	**NF**	**-**		

Total = 3ms, Longest= 2ms

M1: 1,2 M2: 2,3 M3: 3,1

OLAP:

	1ms	2	3	4
M1	NF			
M2	F			
M3	NF			

Total = 3ms, Longest= 1ms

OLTP- edit - Web and Mobile APIs, Apps

-> ACID, Normalising (1st, 2nd, 3rd, BCNF…)

-> best case scenario-> normalised

OLAP- search- DA, ML, Chatbots, Search

-> flat tables

-> Avoid JOINING, Subqueries

-> best case scenario-> ONE BIG TABLE where all our columns are present!

BIG DATA = ML!

Temperatures:

25 26 27 28 27 26 25 26 27 28 —?

-> 29-> 100 days -> 128 deg? -> A1

-> 28 -> A2

-> Avg(n days) -> A3

-> 26.5

-> avg of seasons-> A4

-> Min, Max

temp(d) -> temp(d-1)
->

Today:	**Tomorrow**	**Error (today -p)**
A1: 29	**28.1**	**.9**
A2: 28		**.1**
A3: 26.5		**1.6**
A4: 24.32		
A5: 25		
A6: 28		**.1**
A7: 27.99		

Best algorithms-> A2, A6 and A7

TOO MUCH-> problem, TOO LESS-> problem