Part 2. The notion of **data space** (or feature space) in machine learning

What will be DATA for us in this course?

Data = Table with numbers + object annotation + variable annotation

Variables (features)

Objects (samples, measurements)

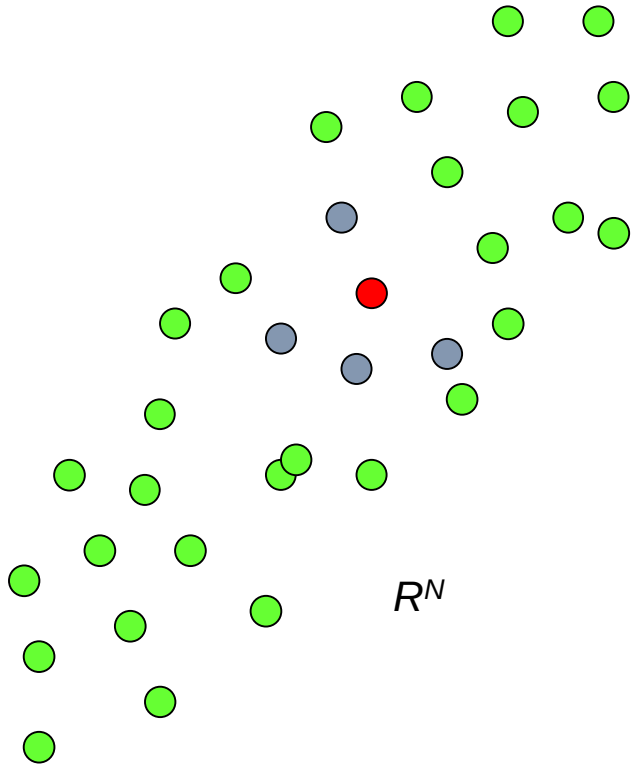
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	ID	GSM26804	GSM26867	GSM26868	GSM26869	GSM26870	GSM26871	GSM26872	GSM26873	GSM26874	GSM26875	GSM26876	GSM26877	GSM26878	GSM26879	GSM26880
2	1007_s_at	10.1865219	8.55465039	10.0171922	9.62855164	8.98179716	9.32096544	9.47013224	8.95127564	9.96641442	10.4723245	9.24634157	9.02814158	9.80726386	10.0884552	9.42789917
3	1053_at	7.14041117	7.9214253	7.19382145	6.33955085	7.0908807	7.14601906	7.11899363	6.1405604	7.07155598	7.54040306	7.13747501	6.68022907	7.3384041	7.06154974	8.10872116
4	117_at	3.82411386	4.04754597	3.79189557	3.84224583	3.92385016	4.86869941	3.88504756	3.76331375	4.32971859	3.89711353	3.81477514	3.86303976	3.75730583	3.90036158	3.7273577
5	121_at	3.61027455	3.54508217	4.54816259	3.74454054	3.61249215	3.92550296	3.6694669	3.52652939	3.64293119	4.04713877	3.46597877	3.49245376	3.67221448	3.66359582	3.61227108
6	1255_g_at	1.88973308	1.83203391	2.04186476	1.89308074	1.91040953	1.91591151	1.95901919	1.83514593	1.91134886	1.98236692	1.89657927	1.91074736	1.9468854	2.00801479	1.87033852
7	1294_at	2.76750098	2.78550183	2.86012235	2.84959436	3.26397282	2.88519676	3.16642211	3.26979855	2.96513014	3.01209778	3.7258176	3.24593083	2.89258523	4.22469552	2.65138576
8	1316_at	3.56186724	6.00938132	5.47627387	3.46082345	3.5589646	3.55022131	3.6495575	3.52929593	3.81489528	3.80151472	3.65353504	3.64297291	5.49390683	3.65494323	3.1776103
9	1320_at	2.73909575	2.68207678	2.97410312	2.73471052	2.78817658	2.79770738	2.90340693	2.67748734	2.78673884	2.94813241	2.74922119	2.78593559	2.88668564	2.98050986	2.62360657
10	1405_i_at	6.56570279	6.28698926	4.91331257	7.08328018	8.85548288	8.73393312	7.00368174	9.20074992	7.56290044	7.08242829	8.62383444	6.68093219	6.64318345	9.43959551	7.59805121
11	1431_at	2.8344133	2.78755371	3.18847354	2.88404293	2.93762587	2.89823055	3.05244607	2.78417436	2.90076657	3.09872342	2.90011368	2.90453628	3.00948297	3.1228764	2.74311179
12	1438_at	2.08209982	2.05046004	2.1380021	2.08249533	2.09277912	2.1099077	2.11854206	2.04375093	2.09150681	2.13821066	2.0847717	2.09495798	2.13115924	2.1353399	2.04584187
13	1487_at	5.54120155	5.35862078	5.46869731	5.52103094	5.51418122	5.55106929	5.4161482	5.44489428	5.24818751	5.56301699	5.42549692	5.54960823	5.82915837	5.56467106	5.50830277
14	1494_f_at	2.54757724	2.37930712	2.62709071	2.38194831	2.44028963	2.4526832	2.4825064	2.4207785	2.60409103	2.49857683	2.43723118	5.2354071	2.48110506	2.49964028	2.41921899
15	1598_g_at	2.7304057	2.67040188	2.59698585	7.93551881	5.34425285	3.13179926	6.57015445	4.4323031	5.18399788	3.88981767	3.85670525	4.88119006	2.70978966	3.85692387	2.75953351
16	160020_at	2.1655937	2.14026455	2.21194547	2.16062823	2.17141169	2.17996571	2.2008294	2.1242019	2.18214481	2.2125988	2.1687426	2.43832316	2.19630922	2.21189546	2.12666118
17	1729_at	7.01826581	6.8620684	6.2748978	5.90084028	6.41997144	6.40378323	6.47535055	6.56605198	6.69687512	6.47743846	6.83935011	6.77296396	7.34317394	6.89120616	6.7314662
18	1773_at	1.65915684	1.63701805	1.72741313	1.65439452	1.67083716	1.67811596	1.70139307	1.64332524	1.67628101	1.71880406	1.6714433	1.67212824	1.70672522	1.71772136	1.6204299
19	177_at	2.94878496	2.86836877	3.14969855	2.97643251	2.98608845	3.03205184	3.08209486	2.89669887	2.97919094	3.13159394	2.92393653	3.02575255	3.12900366	3.1146516	2.95474175
20	179_at	0.57716722	0.55275837	0.63200969	0.57298874	0.58419168	0.59124817	0.61105933	0.56274132	0.59422142	0.62795537	0.58159784	0.58517916	0.61999536	0.61528153	0.5432499
21	1861_at	1.18690202	1.15813312	1.22122377	1.17375236	1.18429212	1.20030196	1.27557097	1.15859558	1.19207924	1.65247824	1.18805205	1.19209823	1.22668581	1.2303746	1.15380531
22	200000_s_at	9.20648723	9.16145477	8.7773438	8.87165851	8.61164901	9.11532903	7.49798068	8.6501605	8.65648402	8.50846148	8.23676007	9.0088335	8.48443715	8.47810052	8.67504714
23	200001_at	10.2111295	9.64241927	8.49184651	9.32048593	9.55080931	9.54725821	9.48348667	9.20829652	9.94634018	9.95504495	9.78220873	9.51833134	10.0545938	9.27885752	9.13860085
24	200002_at	11.7416844	12.5435781	12.5946606	11.2449107	11.7915808	11.4243596	12.3739699	11.5708209	10.6073152	12.4039151	11.1801336	12.3501075	11.8337089	12.0351735	12.0298037
25	200003_s_at	11.9080732	12.7295141	11.8924837	11.8114427	11.9696242	12.0234239	12.1696299	12.4044847	11.5106517	12.6009712	11.214454	13.10743	12.5458678	12.3421479	11.8707809
26	200004_at	12.8626281	13.0318466	12.3226364	12.9112874	12.5629091	13.1340588	13.0250779	12.8029198	12.9787753	13.1286809	12.748781	13.0629905	13.0935061	13.030989	13.4212022
27	200005_at	11.2365327	11.0171526	11.7152353	10.4233686	11.1230332	11.294694	10.7547452	10.900953	10.4631057	10.5860537	10.8269418	10.8355385	11.3292254	10.9910538	11.8222214
28	200006_at	13.4345486	13.07559	13.5937822	13.4856798	13.0994422	13.4686359	13.5762938	13.3161896	13.4856942	13.4639962	13.5249391	13.2203125	13.0822576	13.2736093	13.2935
29	200007_at	13.4323845	13.8222834	13.8399309	13.5619045	12.9873835	13.1472475	13.6921953	13.5192546	13.8453793	14.0467732	13.594668	13.7081125	13.3744476	13.8363235	13.4141853

Geometrical point of view: Analysis of numerical tables = study of a cloud of points in multidimensional space

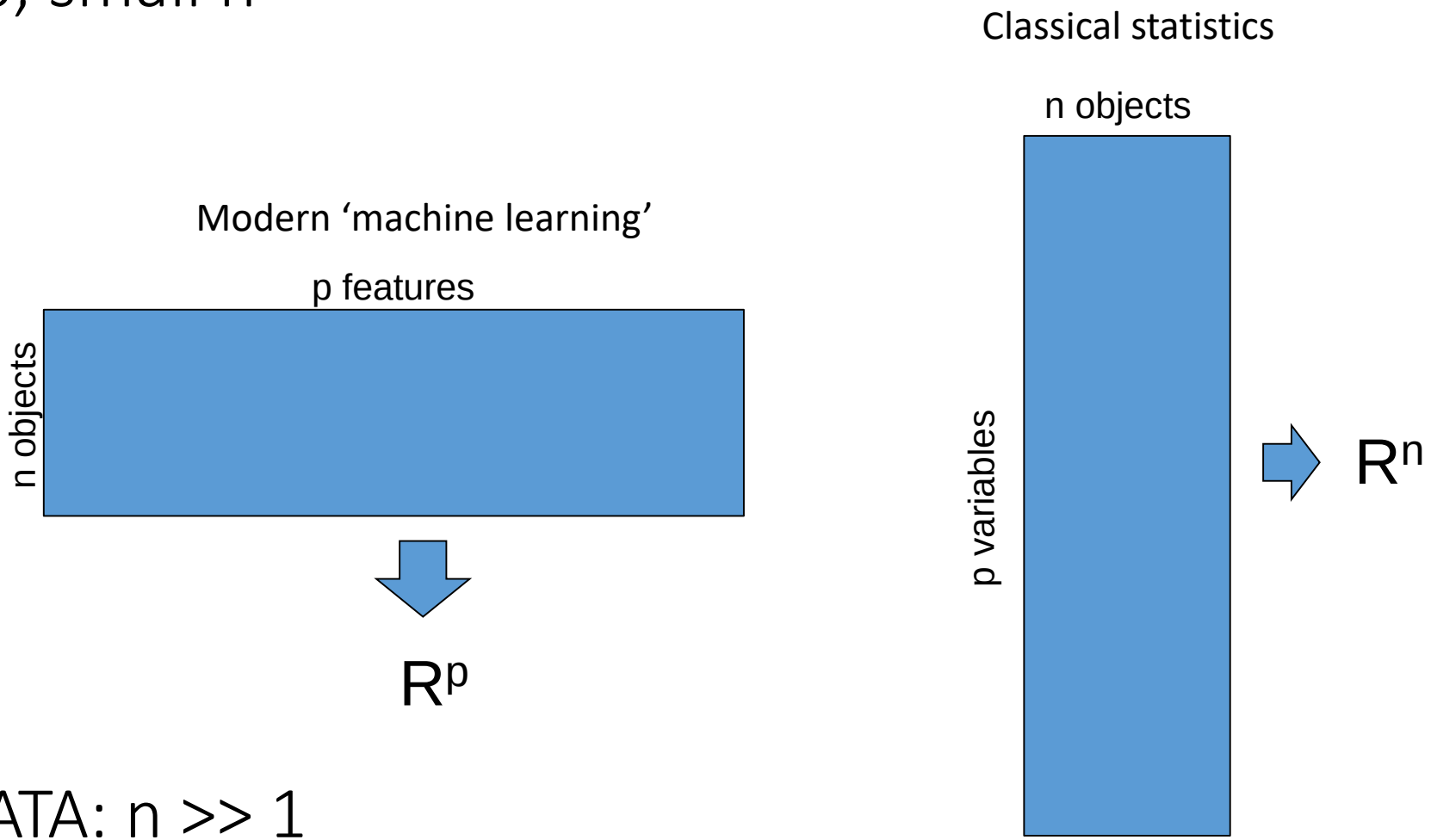
Variables (features)

Objects (samples, measurements)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	ID	GSM26804	GSM26867	GSM26868	GSM26869	GSM26870	GSM26871	GSM26872	GSM26873	GSM26874	GSM26875	GSM26876	GSM26877	GSM26878	GSM26879	GSM26880
2	1007_s_at	10.1865219	8.55465039	10.0171922	9.62855164	8.98179716	9.32096544	9.47013224	8.95127564	9.96641442	10.4723245	9.24634157	9.02814158	9.80726386	10.0884552	9.42789917
3	1053_at	7.14041117	7.9214253	7.19382145	6.33955085	7.0908807	7.14601906	7.11899363	6.1405604	7.07155598	7.54040306	7.13747501	6.68022907	7.3384041	7.06154974	8.10872116
4	117_at	3.82411386	4.04754597	3.79189557	3.84224583	3.92385016	4.86869941	3.88504756	3.76331375	4.32971859	3.89711353	3.81477514	3.86303976	3.75730583	3.90036158	3.7273577
5	121_at	3.61027455	3.54508217	4.54816259	3.74454054	3.61249215	3.92550296	3.6694669	3.52652939	3.64293119	4.04713877	3.46597877	3.49245376	3.67221448	3.66359582	3.61227108
6	1255_g_at	1.88973308	1.83203391	2.04186476	1.89308074	1.91040953	1.91591151	1.95901919	1.83514593	1.91134886	1.98236692	1.89657927	1.91074736	1.9468854	2.00801479	1.87033852
7	1294_at	2.76750098	2.78550183	2.86012235	2.84959436	3.26397282	2.88519676	3.16642211	3.26979855	2.96513014	3.01209778	3.7258176	3.24593083	2.89258523	4.22469552	2.65138576
8	1316_at	3.56186724	6.00938132	5.47627387	3.46082345	3.5589646	3.55022131	3.6495575	3.52929593	3.81489528	3.80151472	3.65353504	3.64297291	5.49390683	3.65494323	3.1776103
9	1320_at	2.73909575	2.68207678	2.97410312	2.73471052	2.78817658	2.79770738	2.90340693	2.67748734	2.78673884	2.94813241	2.74922119	2.78593559	2.88668564	2.98050986	2.62360657
10	1405_i_at	6.56570279	6.28698926	4.91331257	7.08328018	8.55548288	8.73393312	7.00368174	9.20074992	7.56290044	7.08242829	8.62383444	6.68093219	6.64318345	9.43959551	7.59805121
11	1431_at	2.8344133	2.78755371	3.18847354	2.88404293	2.93762587	2.89823055	3.05244607	2.78417436	2.90076657	3.09872342	2.90011368	2.90453628	3.00948297	3.1228764	2.74311179
12	1438_at	2.08209982	2.05046004	2.1380021	2.08249533	2.09277912	2.1099077	2.11854206	2.04375093	2.09150681	2.13821066	2.0847717	2.09495798	2.13115924	2.1353399	2.04584187
13	1487_at	5.54120155	5.35862078	5.46869731	5.52103094	5.51418122	5.55106929	5.4161482	5.44489428	5.24818751	5.56301699	5.42549692	5.54960823	5.82915837	5.56467106	5.50830277
14	1494_f_at	2.54757724	2.37930712	2.62709071	2.38194831	2.44028963	2.4526832	2.4825064	2.4207785	2.60409103	2.49857683	2.43723118	5.2354071	2.48110506	2.49964028	2.41921899
15	1598_g_at	2.7304057	2.67040188	2.59698585	7.93551881	5.34425285	3.13179926	6.57015445	4.4323031	5.18399788	3.88981767	3.85670525	4.88119006	2.70978966	3.85692387	2.75953351
16	160020_at	2.1655937	2.14026455	2.21194547	2.16062823	2.17141169	2.17996571	2.2008294	2.1242019	2.18214481	2.2125988	2.1687426	2.43832316	2.19630922	2.21189546	2.12666118
17	1729_at	7.01826581	6.8620684	6.2748978	5.90084028	6.41997144	6.40378323	6.47535055	6.56605198	6.69687512	6.47743846	6.83935011	6.77296396	7.34317394	6.89120616	6.7314662
18	1773_at	1.65915684	1.63701805	1.72741313	1.65439452	1.67083716	1.67811596	1.70139307	1.64332524	1.67628101	1.71880406	1.6714433	1.67212824	1.70672522	1.71772136	1.6204299
19	177_at	2.94878496	2.86836877	3.14969855	2.97643251	2.98608845	3.03205184	3.08209486	2.89669887	2.97919094	3.13159394	2.92393653	3.02575255	3.12900366	3.1146516	2.95474175
20	179_at	0.57716722	0.55275837	0.63200969	0.57298874	0.58419168	0.59124817	0.61105933	0.56274132	0.59422142	0.62795537	0.58159784	0.58517916	0.61999536	0.61528153	0.5432499
21	1861_at	1.18690202	1.15813312	1.22122377	1.17375236	1.18429212	1.20030196	1.27557097	1.15859558	1.19207924	1.65247824	1.18805205	1.19209823	1.22668581	1.2303746	1.15380531
22	200000_s_at	9.20648723	9.16145477	8.7773438	8.87165851	8.61164901	9.11532903	7.49798068	8.6501605	8.65648402	8.50846148	8.23676007	9.0088335	8.48443715	8.47810052	8.67504714
23	200001_at	10.2111295	9.64241927	8.49184651	9.32048593	9.55080931	9.54725821	9.48348667	9.20829652	9.94634018	9.95504495	9.78220873	9.51833134	10.0545938	9.27885752	9.13860085
24	200002_at	11.7416844	12.5435781	12.5946606	11.2449107	11.7915808	11.4243596	12.3739699	11.5708209	10.6073152	12.4039151	11.1801336	12.3501075	11.8337089	12.0351735	12.0298037
25	200003_s_at	11.9080732	12.7295141	11.8924837	11.8114427	11.9696242	12.0234239	12.1696299	12.4044847	11.5106517	12.6009712	11.214454	13.10743	12.5458678	12.3421479	11.8707809
26	200004_at	12.8626281	13.0318466	12.3226364	12.9112874	12.5629091	13.1340588	13.0250779	12.8029198	12.9787753	13.1286809	12.748781	13.0629905	13.0935061	13.030989	13.4212022
27	200005_at	11.2365327	11.0171526	11.7152353	10.4233686	11.1230332	11.294694	10.7547452	10.900953	10.4631057	10.5860537	10.8269418	10.8355385	11.3292254	10.9910538	11.8222214
28	200006_at	13.4345486	13.07559	13.5937822	13.4856798	13.0994422	13.4686359	13.5762938	13.3161896	13.4856942	13.4639962	13.5249391	13.2203125	13.0822576	13.2736093	13.2935
29	200007_at	13.4323845	13.8222834	13.8399309	13.5619045	12.9873835	13.1472475	13.6921953	13.5192546	13.8453793	14.0467732	13.594668	13.7081125	13.3744476	13.8363235	13.4141853

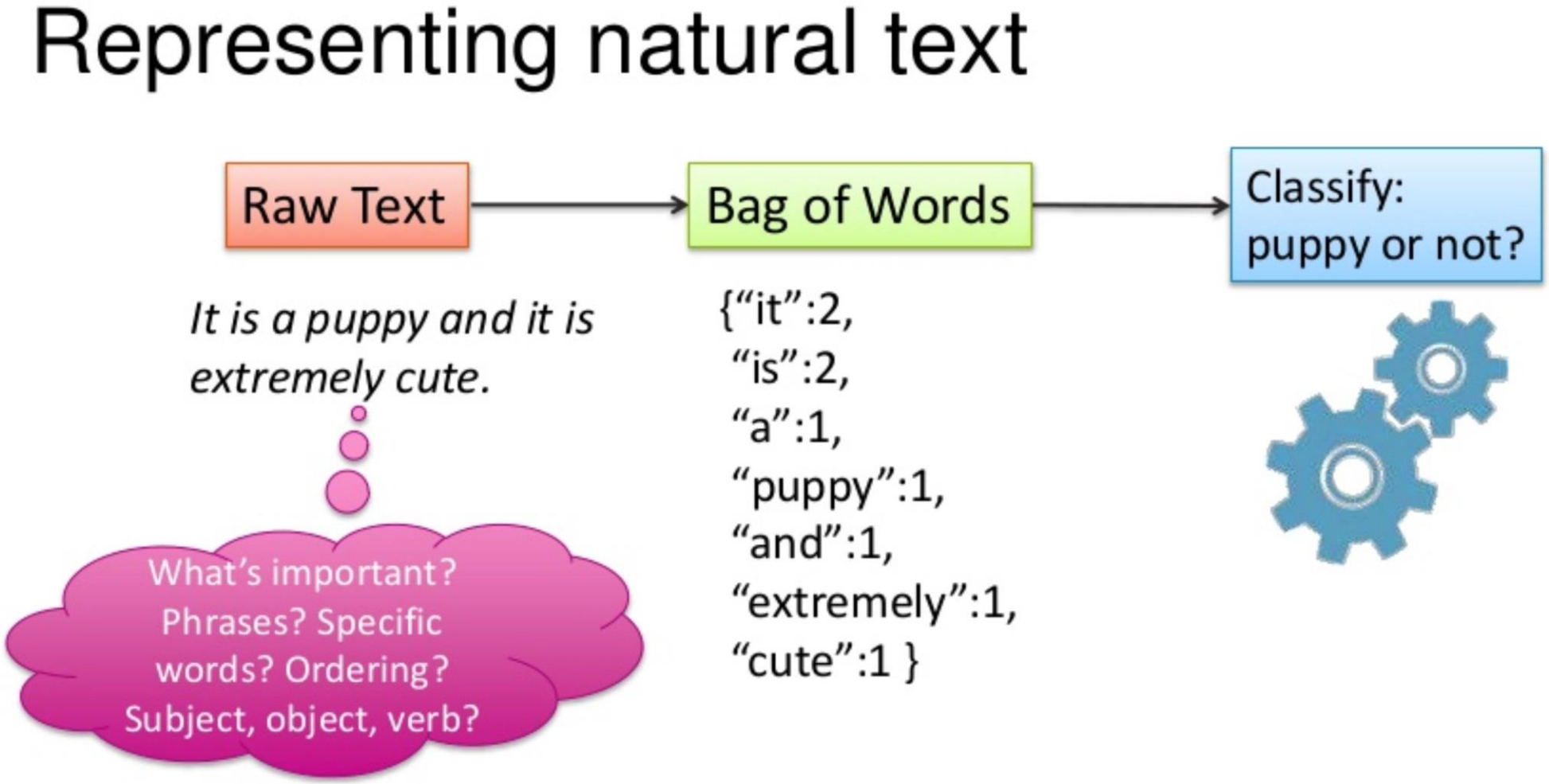


Large p, small n

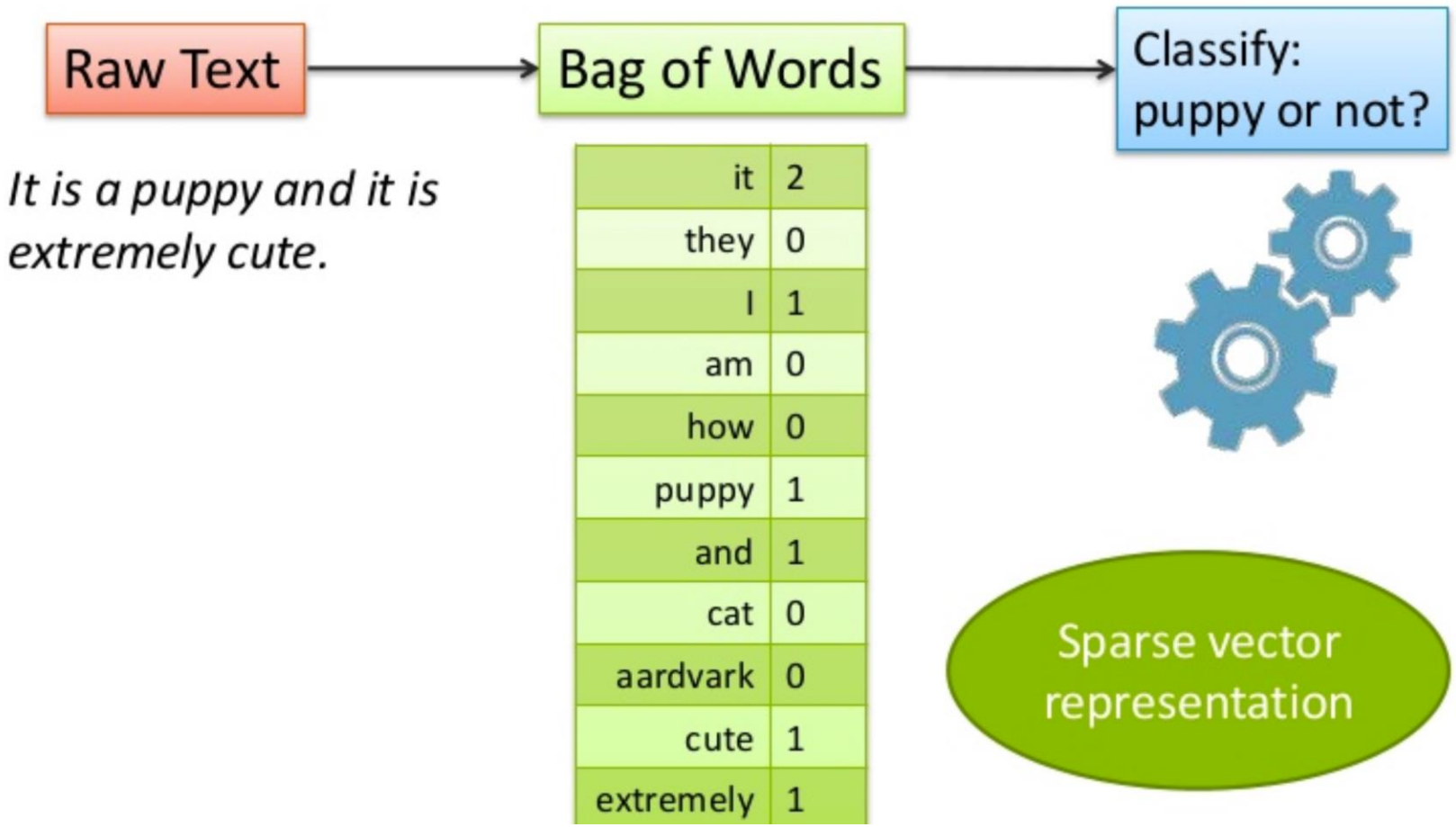


BIG DATA: n >> 1
WIDE DATA: p>>n
REAL-WORLD BIG DATA: p>>n>>1(most frequently)

Other data types: raw data -> numerical table



Representing natural text



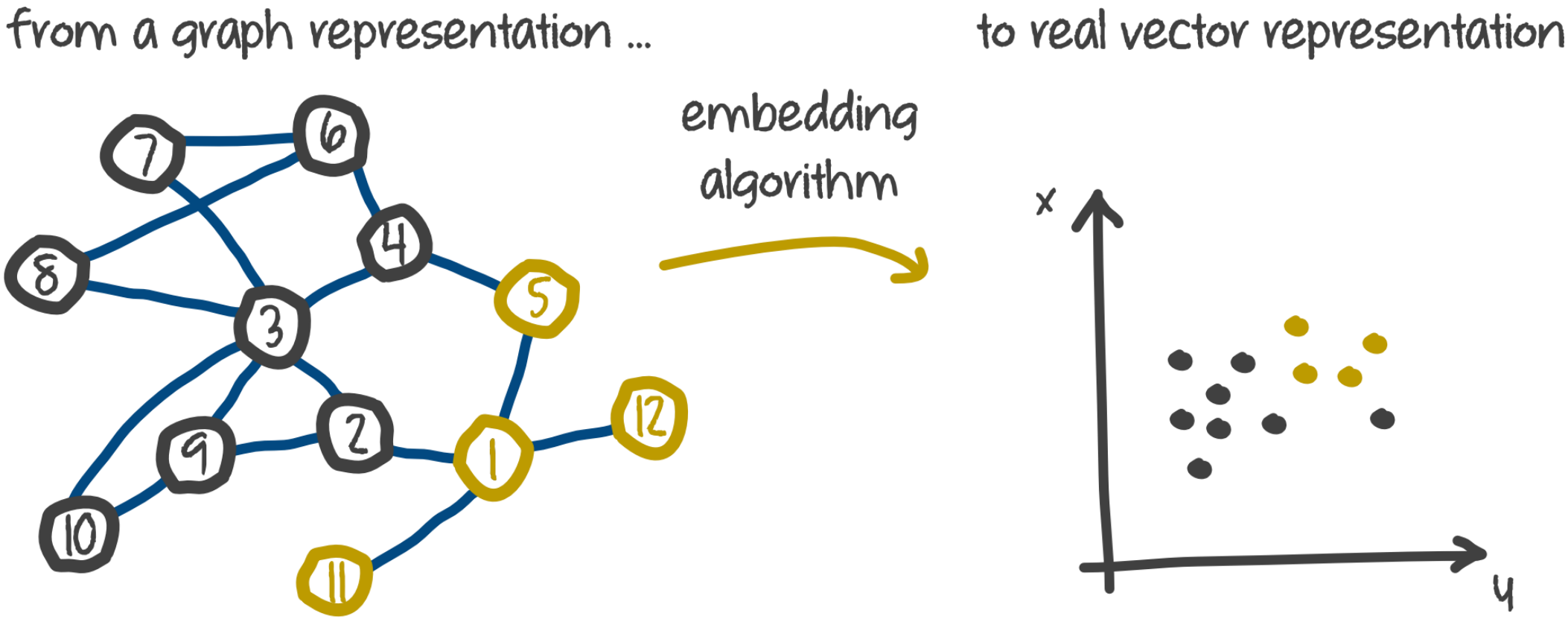
Representing images

Raw Image → Bag of Visual Words → Classify: person or animal?

Raw image:
millions of RGB triplets,
one for each pixel

Image source: "Recognizing and learning object categories,"
Li Fei-Fei, Rob Fergus, Anthony Torralba, ICCV 2005—2009.

Graph embedding



Example: recommendation systems

Data types: most of the world data are not numbers!

1) Numerical
   - Example: *weight, height*
2) Categorical:
   - Ordinal
     - Example: age range (infant, toddler, teenager, young, adult, senior)
   - Nominal
     - Example: *eye color, mothertongue*

Simplest data type: BINARY! (Yes/No, False/True, 0/1)

Data types: Numerical

Example: *weight, height*

Must be normalized (made comparable)!

Simplest normalization

z-score: subtract the mean, divide by standard deviation

taking log: by itself make the numbers more comparable

The appropriate normalization depends on the initial (raw) distribution (histogram)

The final distribution (after normalization) can be a hyperparameter of supervised learning

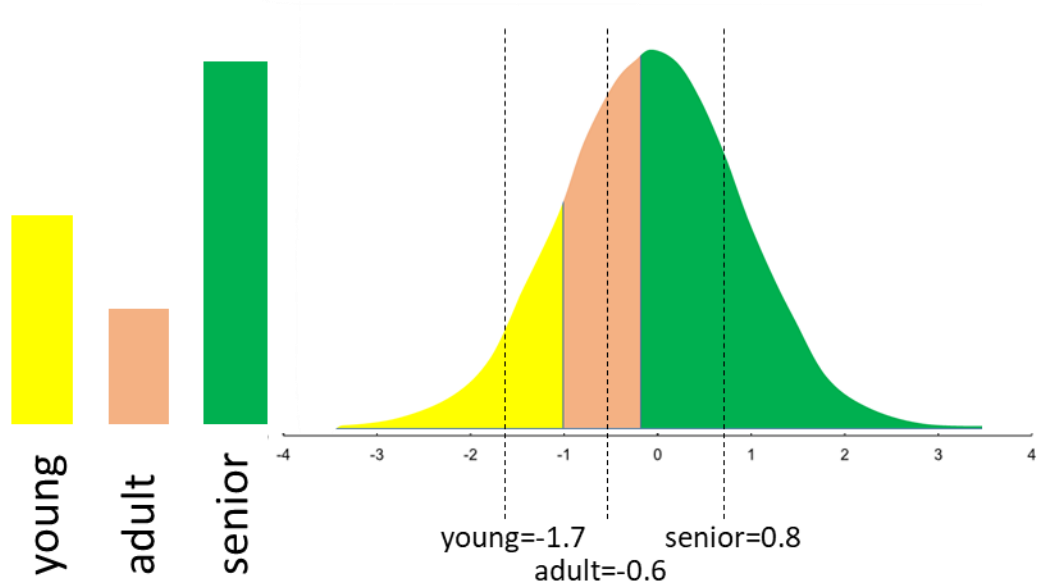
Data types: Categorical, ordinal

Example: age range (infant, toddler, teenager, young, adult, senior)

Must be quantified : methods for ordinal variable quantification, univariate and multivariate

Simplest univariate: act if the ordinal value is a discretization of a normal distribution

Simplest multivariate: maximize the correlation between all quantified ordinal variables, and between all ordinal and numerical variables



Data types: Categorical, nominal

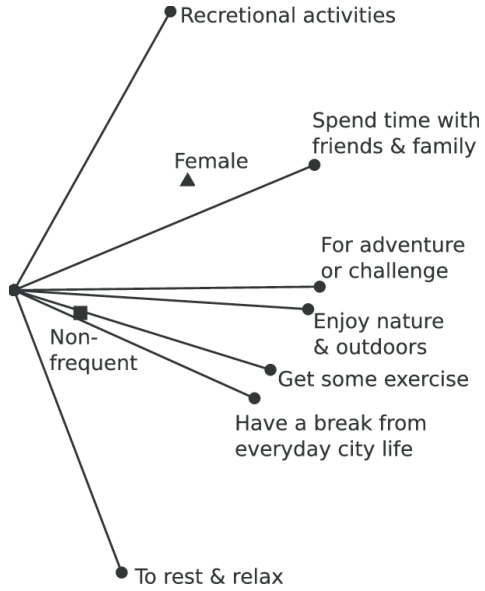
Example: *eye color*

Must be converted to numbers

Simplest encoding: dummy encoding

Eye color		Eye color: BLACK	Eye color: BLUE	Eye color: BROWN	Eye color: GREEN
BLACK		1	0	0	0
BLUE		0	1	0	0
BROWN		0	0	1	0
GREEN		0	0	0	1
GREEN		0	0	0	1
BROWN		0	0	1	0
BLUE		0	1	0	0

More sophisticated approach: **CatPCA**



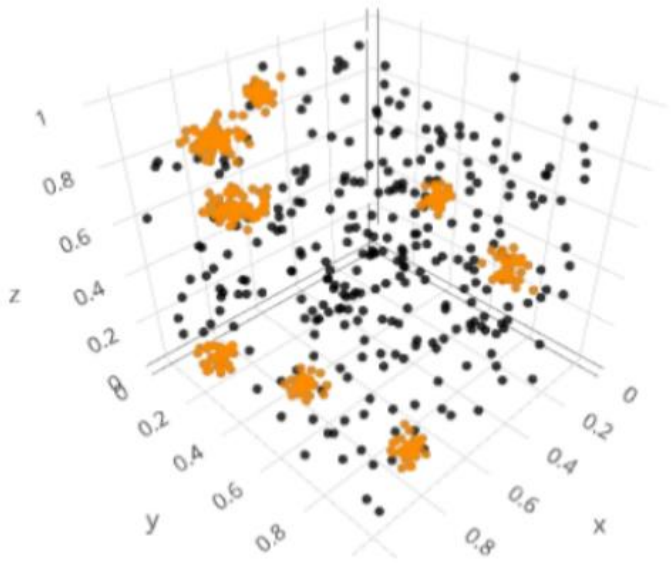
Data types: small conclusion

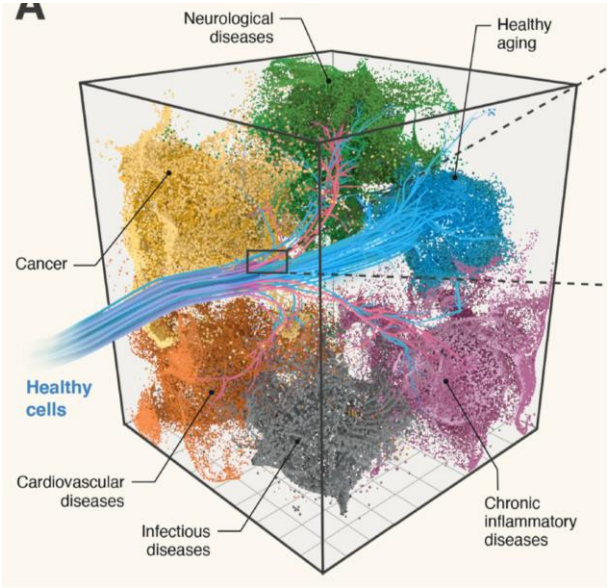
Quantification of data affects all aspects of machine learning and AI, being the most fundamental hyperparameter of any method

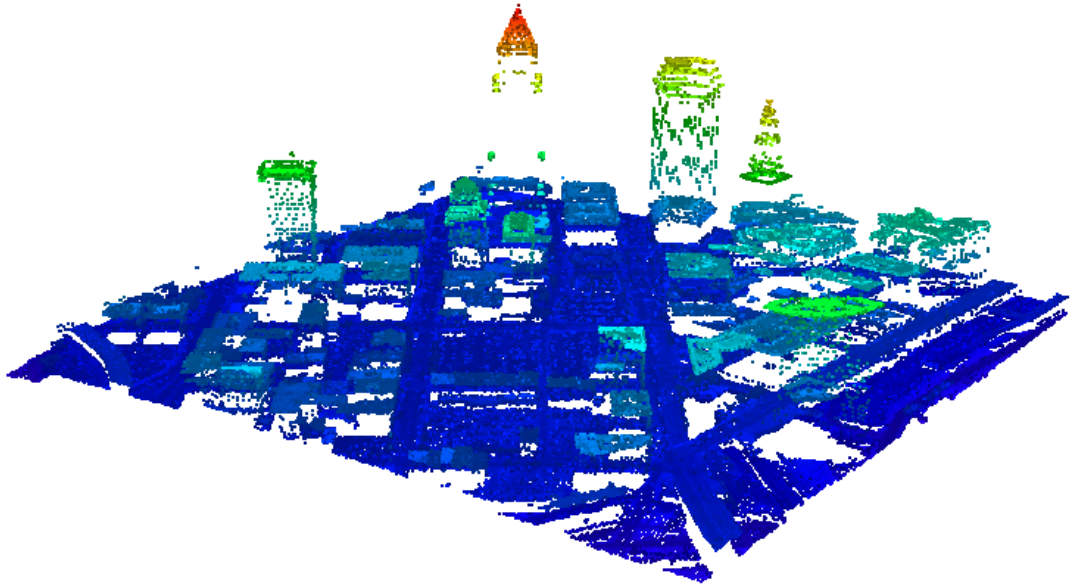
Quantification of data is tightly related to the definition of distance (next section in this lecture)

Quantification of data is a subject of unsupervised learning by itself: normalization of numerical data (learning the target distribution), ordinal (optimal scaling), nominal (CatPCA)

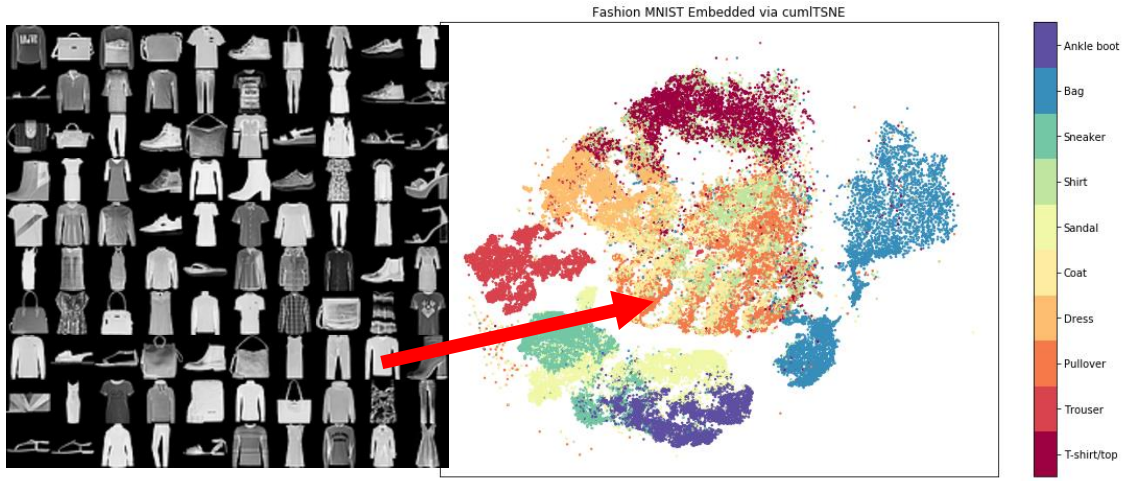
Data point cloud in R^N

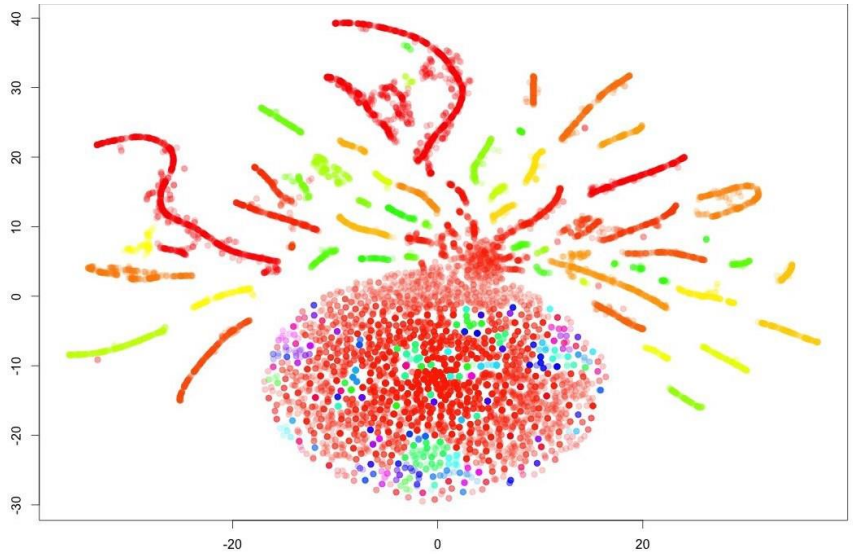






LIDAR Data point cloud





Augmented feature space

One can add to the original features, a set of arbitrary functions of them, i.e., all pairwise products

If one can guess the right set of basis functions for data augmentation (e.g., polynomial basis of small degree), then the new features can be generated using this basis

One of the most popular basis is the basis of radial functions

Augmented feature space can be used for learning, and some non-linear problems can become linear in the augmented space

Augmenting feature space can be made implicit (without adding new columns in the table), this is the idea of *kernel trick*

Kernel trick in two words

Gramm matrix is the matrix of scalar products

Many classical machine learning algorithms can be written down only using the Gramm matrix

Kernel trick consists in substituting the Gramm matrix with Kernel matrix, which is a Gramm matrix computed in some augmented feature space (sometimes infinite-dimensional!) and act as it would be the actual Gramm matrix

$$\boldsymbol{\phi\phi^T} = K = \begin{bmatrix} \phi(x_1)^T\phi(x_1) & \phi(x_1)^T\phi(x_2) & . & \phi(x_1)^T\phi(x_n) \\ \phi(x_2)^T\phi(x_1) & \phi(x_2)^T\phi(x_2) & . & . \\ . & . & . & . \\ \phi(x_m)^T\phi(x_1) & \phi(x_m)^T\phi(x_2) & . & \phi(x_m)^T\phi(x_n) \end{bmatrix} = \begin{bmatrix} k(x_1,x_1) & k(x_1,x_2) & . & k(x_1,x_n) \\ k(x_2,x_1) & k(x_2,x_2) & . & . \\ . & . & . & . \\ k(x_m,x_1) & k(x_m,x_2) & . & k(x_m,x_n) \end{bmatrix}$$

Kernel trick is a powerful way of making classical linear statistical methods (linear regression, principal component analysis) applicable to non-linear data structure