Machine learning with mismatched datasets

Using a meta-model to compensate for datasets with different distributions

Introduction

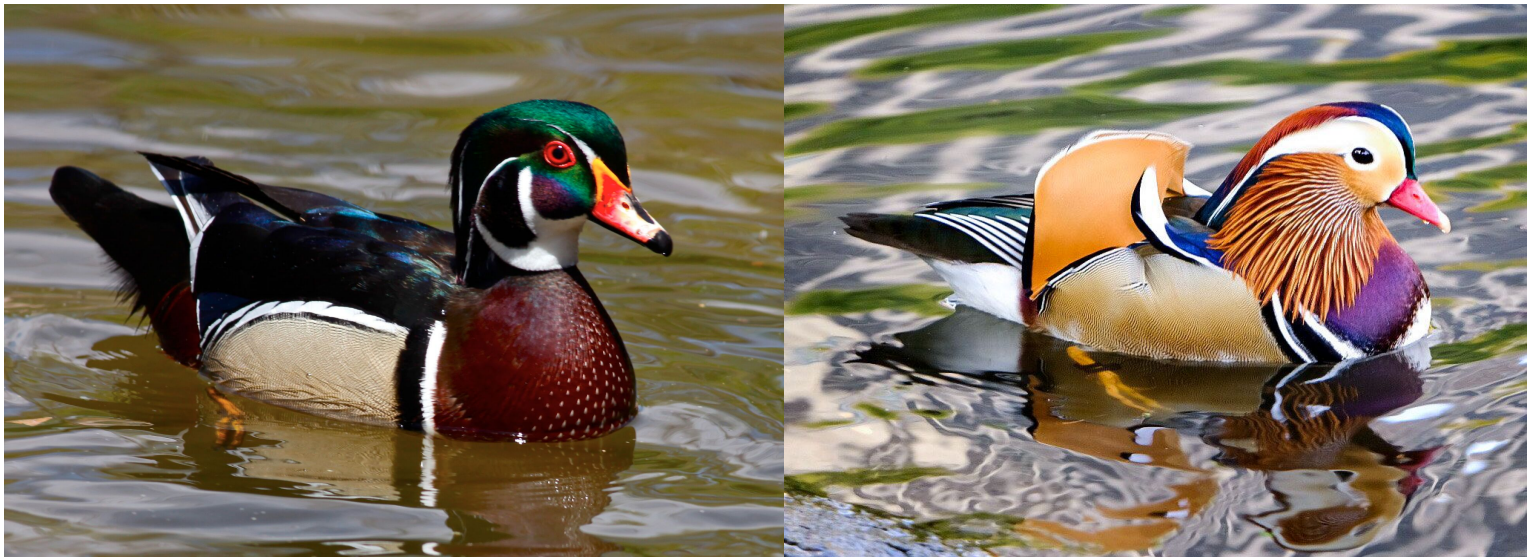
Dealing with datasets that are similarly distributed

$$\Pr[A_i|E] = \frac{\Pr[E|A_i]\Pr[A_i]}{\sum_{i=1}^{n}\Pr[E|A_i]\Pr[A_i]},$$

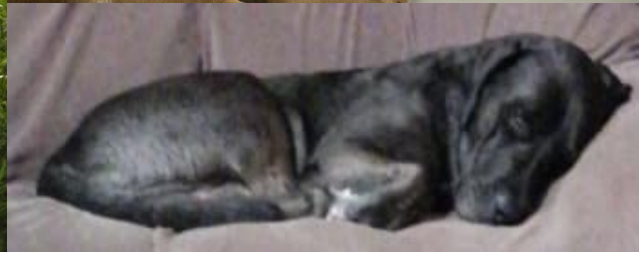
Duck age detection

1. A dog dataset with ages
2. A cat dataset with ages
3. A bird dataset with ages

Ducks



Dogs



Cats


PLEASE DON'T
LET THE CAT OUT.
NO MATTER
WHAT IT
TELLS YOU!
HELP
Cat needs
air

Birds

So what to pick

So what to pick

Birds, but why?

1. The ducks and birds look similar to me
2. A dataset of birds could contain ducks
3. Ducks could share features with different types of birds, and it is unlikely to share aging features with cats or dogs

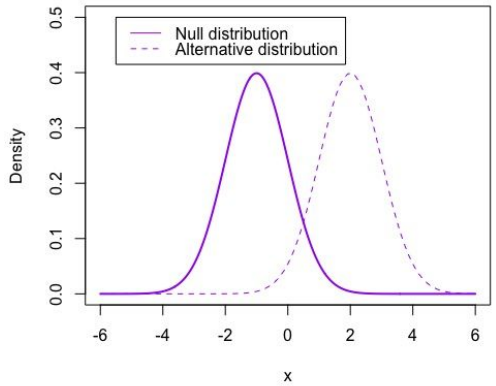
So why do we need an architecture for this

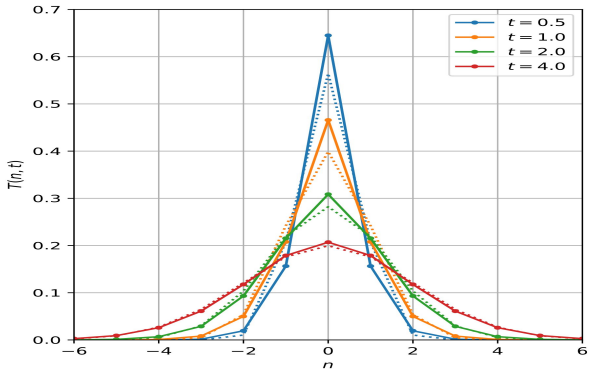
So why do we need an architecture for this

0	1	2	3	target
25.851100235128698	795.7163107045812	0.050057187408672446	3.0116628631591986	1.0113529582980931
12.337794540855652	282.7685803887259	0.14313010568883544	3.2278036352152384	1.268777779071366
24.838373711533496	647.4582153627362	0.2595972572016474	4.926097501983797	1.4187976936775826
15.222612486575871	924.6038097971995	0.06369379659896308	4.8523375508920115	1.3071166835145265
25.865240118356347	663.6908389138603	0.12019346929761689	2.490507445424394	1.2659016444003024
45.03722843377683	998.2348129255939	0.2067120890796214	4.96161307834657	1.3328912266717288
48.81945761480191	938.0724428351286	0.09252210568488896	1.6952739161644117	1.0758631666871779
13.491520978228444	924.6242849074127	0.09917341691652505	3.6055381250252605	1.417102156530998
52.8944765075251	642.8420321662145	0.39593855697523667	3.0775781550303147	1.3690871277214822
39.32504638407918	889.0791237627866	0.059144138672095906	5.250721574724837	0.9261740862508275
54.443054445324734	818.4573632385813	0.19022199603220258	5.446396642257443	1.2490165581109307
10.1613003288821	573.1908581073357	0.5042977515465478	2.9680707418683974	1.515056268461215
19.388766929317434	313.55428493704983	0.05968347893514854	4.894177664699455	0.7644495923912876
15.581405800002951	424.24736804689155	0.29578657964016913	1.7668127255854018	1.443418960893867
33.70588027460065	327.1950825158325	0.3446527684516421	4.998791800104655	1.2916907661599035
10.116721441391292	545.5518800092196	0.39720007886387254	3.570896347634513	1.5131443475192259

Why mismatched datasets are a problem

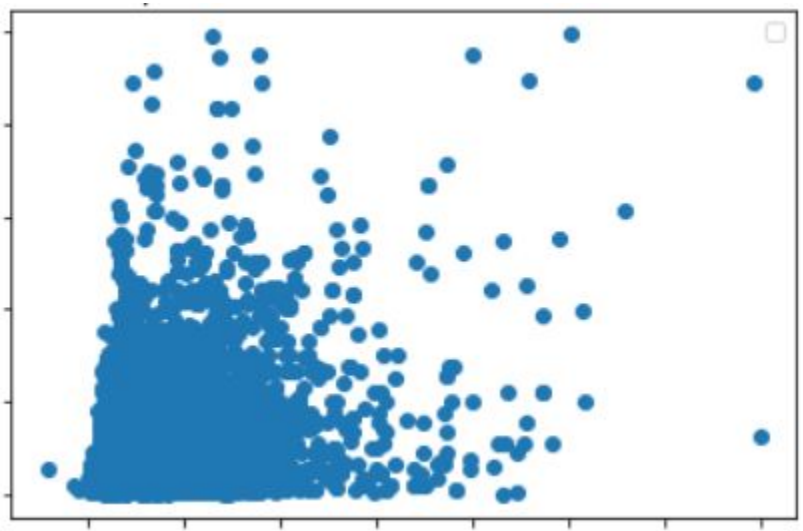
- Bayesian models work on similarity



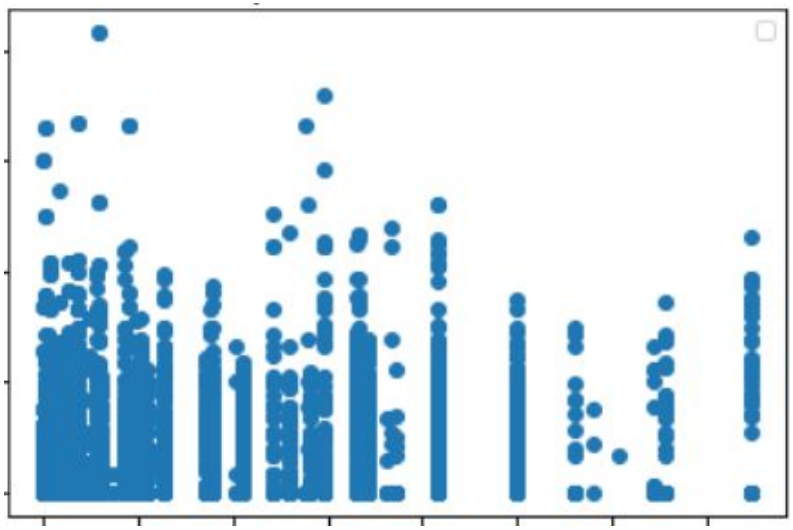


How the mismatch presents itself

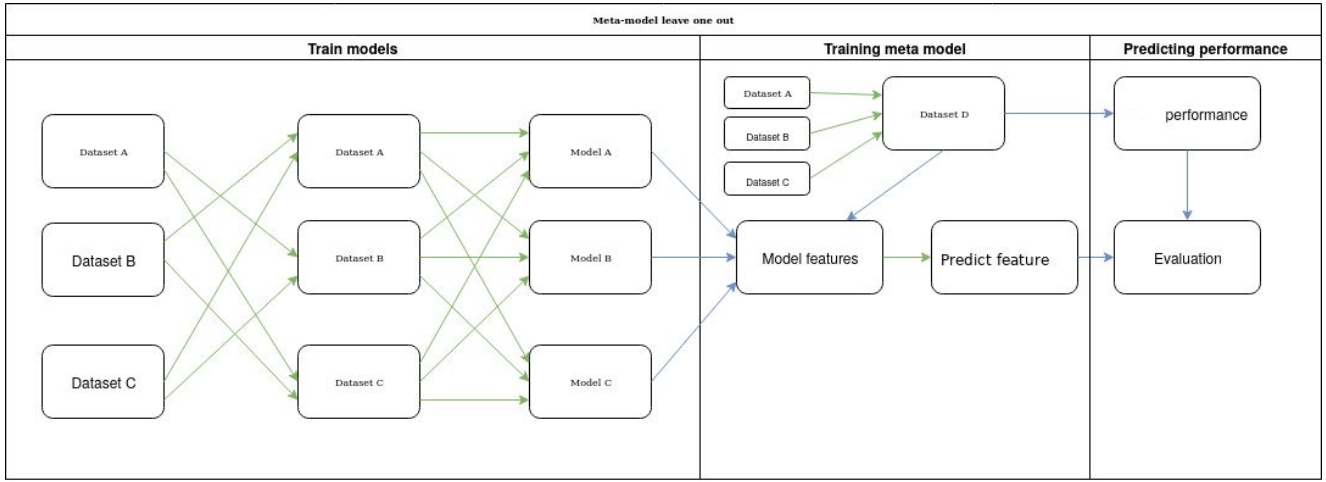
Regression performance same distribution



Regression performance different distributions



Meta-model approach



Generating regression data

1. For each feature, we uniformly generated a value between 0 and 1. For each dataset, we generated the features with the seed equal to the number of the dataset (e.g. dataset 0 generates on seed 0, dataset 1 generates on seed 1 etc.)

2. Then we applied the following transformations to each feature

   - Feature 1: $x_1 \times 100$
   - Feature 2: $x_2 \times 520\pi + 40\pi$
   - Feature 3: x_3
   - Feature 4: $x_4 \times 10 + 1$

3. We finally applied the following transformation to get the target for each element: $target = tan^{-1}(\frac{x_2 x_3 - \frac{1}{x_2 x_4}}{x_1})$

Related datasets

- Regression performance
- How well can it identify which datasets to use
- Which changes can the meta-model adjust to

Measurements

Correlation coefficient

Kendall Tau

Measured by creating an original dataset and 10 altered datasets, using leave one out cross validation

Modifications investigated

- Noise: Applying random changes to the target of each dataset.
- Scale changes: Changing the range of the input features.
- Data shifts: Changing the relation of datasets i.e. $P_{source}(x,y) \neq P_{target}(x,y)$
- A combination of scale changes and data shifts.
- Model selection: Testing how the model performs with different underlying regression algorithms.
- Dataset size: Testing out various different sizes of datasets.

Additive noise

$$tan^{-1}(\frac{x_2x_3 - \frac{1}{x_2x_4}}{x_1}) + N(0,1) \times c \times dataset\ number$$

Additive noise

Noise	0	0.01	0.02	0.05	0.1
Metric	coefficients	coefficients	coefficients	coefficients	coefficients
Average	0.9853	0.9855	0.9798	0.9616	0.9663
Standard deviation	0.0143	0.0152	0.0218	0.0454	0.0357
Median	0.9926	0.9921	0.9912	0.9819	0.9845
Metric	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau
Average	0.9273	0.9232	0.9192	0.8949	0.8505
Standard deviation	0.0637	0.0491	0.0623	0.0802	0.0999
Median	0.9111	0.9111	0.9111	0.9111	0.8667

Scale changes

- We scaled each of the features generated in step 1 to create several unique datasets.
  - For example, if the scale change was five and there are five datasets, the datasets would have the following ranges for the features [(0,0.2), (0.2,0.4), (0.4,0.6), (0.6,0.8), (0.8,1)].
- Here we are primarily testing for whether the meta-model can identify which datasets are relatively close to each target dataset in terms of scale changes.

Scale changes

Scale changes	10	20	50	100
Metric	coefficients	coefficients	coefficients	coefficients
Average	0.9827	0.9712	0.9529	0.9454
Standard deviation	0.0321	0.0466	0.0881	0.1002
Median	0.9981	0.9988	0.9983	0.9983
Metric	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau
Average	0.9515	0.9232	0.8828	0.8949
Standard deviation	0.0285	0.0797	0.1164	0.1094
Median	0.9667	0.9556	0.9111	0.9111

Data shift

$$target = tan^{-1}(\frac{x_2x_3 - \frac{1}{x_2x_4}}{x_1}) + c \times dataset\ number$$

Data shift

Data shift	0.01	0.02	0.05	0.1
Metric	coefficients	coefficients	coefficients	coefficients
Average	0.4478	0.5043	0.5987	0.6082
Standard deviation	0.6096	0.5610	0.4201	0.3957
Median	0.7372	0.7081	0.8010	0.7560
Metric	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau
Average	0.3778	0.3818	0.4869	0.4626
Standard deviation	0.5440	0.5123	0.3120	0.3428
Median	0.4667	0.5556	0.6000	0.6000

Combining data shifts and scale changes

Section	Scale changes	10	20	50
	Metric	coefficients	coefficients	coefficients
Scale changes	Average	0.9785	0.9752	0.9695
	Standard deviation	0.0191	0.0203	0.0395
	Median	0.9862	0.9829	0.9831
	Metric	Kendall Tau	Kendall Tau	Kendall Tau
Scale changes	Average	0.8459	0.8286	0.8277
	Standard deviation	0.0385	0.0664	0.0962
	Median	0.8381	0.8476	0.8476
	Metric	coefficients	coefficients	coefficients
Data shifts	Average	0.7416	0.7054	0.7621
	Standard deviation	0.2924	0.4539	0.3241
	Median	0.8873	0.9094	0.8869
	Metric	Kendall Tau	Kendall Tau	Kendall Tau
Data shifts	Average	0.5749	0.5697	0.5905
	Standard deviation	0.0621	0.1166	0.0853
	Median	0.5810	0.6381	0.6000

Regression algorithm selection

Regression algorithm name	catboost	Light GBM	XGBoost	GBT	Random forest	Extra tree	all regression algorithms
Metric	coefficients	coefficients	coefficients	coefficients	coefficients	coefficients	coefficients
Average	0.9863	0.9886	0.9903	0.9594	0.9874	0.9860	0.9748
Standard deviation	0.0146	0.0119	0.0067	0.0449	0.0119	0.0132	0.0202
Median	0.9930	0.9934	0.9912	0.9833	0.9938	0.9880	0.9805
Metric	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau
Average	0.8909	0.8990	0.9071	0.8343	0.9394	0.9030	0.8454
Standard deviation	0.0608	0.0565	0.0464	0.1072	0.0850	0.0623	0.0542
Median	0.8667	0.8667	0.9111	0.8222	0.9556	0.9111	0.8441

Dataset size

Dataset size	10	100	1000	10000
Metric	coefficients	coefficients	coefficients	coefficients
Average	0.8363	0.9261	0.9817	0.9857
Standard deviation	0.2848	0.1678	0.0217	0.0130
Median	0.9879	0.9923	0.9934	0.9907
Metric	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau
Average	0.6929	0.8101	0.9515	0.9475
Standard deviation	0.1233	0.1688	0.0464	0.0556
Median	0.7333	0.8222	0.9556	0.9556

Conclusion

As long as the relationship between features and targets is preserved, the meta-model is able to predict the MAE with a fairly high amount of accuracy.

The meta-model provides a good baseline for which datasets to use to train models in the absence of additional information about how the datasets are related to a target dataset.

Questions?

Questions

1. Where could you potentially apply this?
2. How would you alter this to be beneficial to datasets that are similarly distributed?
3. Why wasn't this implemented earlier?

Related work

- Distance metrics
- Using a portion of the target dataset
- Robustness