Generating biased data with GANs (UnfairGAN)

Elias Baumann

09 2019

Introduction

- Data contains and reflects real world bias
- E.g. Chile college admissions
- Research needs (biased) data
- Publishing sensitive data is problematic

Research Goal

- Generate Data with GAN
- Prove data has same distribution
- Prove data shows same bias
- Ensure its difficult to abuse data (privacy aspect)

Methodology

- Wasserstein GAN to generate new artificial data
- 4 Datasets:
  - Chile
  - Schufa
  - SQF
  - COMPAS
- Evaluation :
  - Distribution comparison: Model comparison
  - Bias reproduction: Aequitas
  - Privacy: Predict protected attributes
  - Everything: 5-Fold stratified cross validation

Results - Variable Importance comparison

- Extra Trees Variable importance differences (real vs gen.)

Dataset	MAD	Min	Max	stdev	Spearman corr. (p)
Chile	0.00411	0.000049	0.108366	0.016147	0.954 (9.36E-37)
SQF	0.01453	0.000017	0.100716	0.020177	0.690 (5.66E-7)
Compas	0.02671	0.000528	0.127128	0.037223	0.885 (1.77E-4)
Schufa	0.00296	0.000047	0.023399	0.004754	0.899 (2.92E-18)

Results - Running different models to predict, compare metrics

- Chile dataset, predicting admission

	real			generated		
GLM	pos	neg	Accuracy	pos	neg	Accuracy
pred. pos	6038.2	11333	0.6667	5406.2	10398.2	0.733
pred. neg	6069.6	28775.4		3544.2	32867.6	
xgboost						
pred. pos	5569	11802.2	0.6764	4243.2	11561.2	0.7316
pred. neg	5096.2	29748.8		2451.8	33960	

Results - AUC Chile GLM

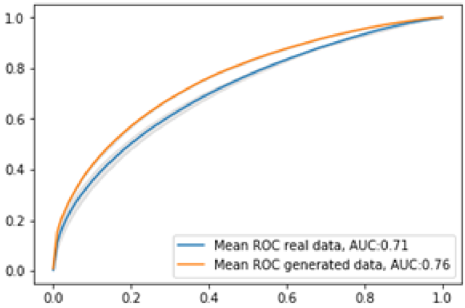

Figure: Predicting university admissions with GLM on gen. vs real data

Aequitas results - Mean difference between audits (with stdev)

Chile	disp._impact	dem._parity	fpr_parity	fnr_parity	ppv_parity	npv_parity	accuracy_parity
nationality	0.00+-0.00	**0.69+-0.39**	**0.58+-0.27**	**-1.68+-0.70**	0.00+-0.00	0.00+-0.00	0.00+-0.00
	0.61+-0.76	0.40+-0.34	0.37+-0.30	**-3.34+-1.08**	0.08+-0.14	**-1.26+-0.59**	-0.27+-0.29
gender	0.00+-0.00	0.16+-0.40	0.09+-0.03	-0.38+-0.10	0.00+-0.00	0.00+-0.00	0.00+-0.00
	0.07+-0.04	0.12+-0.39	0.09+-0.03	**-0.74+-0.21**	0.04+-0.02	-0.21+-0.09	-0.03+-0.04
region	0.00+-0.00	0.13+-0.38	0.06+-0.03	-0.23+-0.11	0.00+-0.00	0.00+-0.00	0.00+-0.00
	0.02+-0.01	0.12+-0.39	0.08+-0.06	**-0.45+-0.42**	0.02+-0.02	-0.13+-0.15	0.01+-0.01
income	0.00+-0.00	0.11+-0.42	0.07+-0.04	-0.15+-0.09	0.00+-0.00	0.00+-0.00	0.00+-0.00
	0.13+-0.36	0.11+-0.38	0.12+-0.20	-0.13+-0.48	0.14+-0.27	0.02+-0.22	0.04+-0.01
	0.34+-0.86	0.08+-0.27	0.12+-0.19	0.11+-0.42	0.30+-0.63	0.06+-0.17	0.06+-0.06

Protected variable prediction (chile, gender)

1. training and predicting protected variables within real and generated data
2. training on generated, predicting on real

GLM: gender	fpr	tpr	auc
real	0.4161	0.5839	0.6648
gen	0.4018	0.5982	0.6923
gen_real	0.4305	0.5694	0.6380

Protected variable prediction (chile, gender)

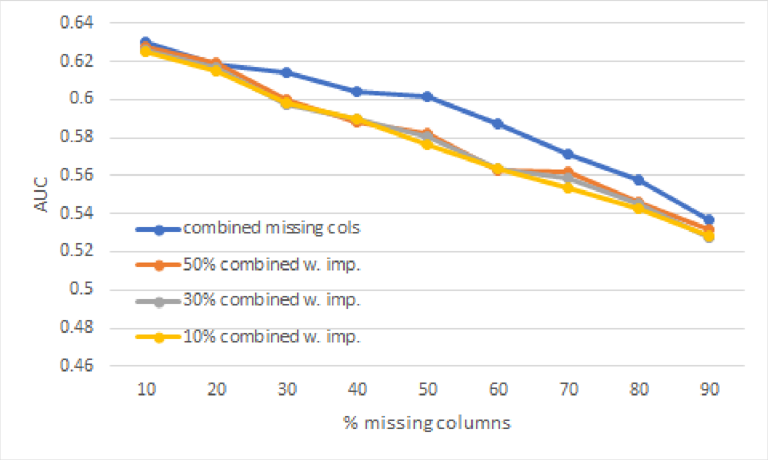

Figure: Mean AUC over CV predicting gender using different data compositions. Imputation via iterative xgboost predictions.

Discussion

- Results:
  - Limited reproduction of data as well as bias
  - Rare cases are missing
  - Generated variables are more correlated
  - Generated data can be used to predict parts of real data
- Potential TODOs:
  - Stratified K-fold only accounts for 1 variable
  - Account for balance
  - Improve performance for smaller datasets
  - tuning?
  - inject additional randomness (Karras et al. 2018)
  - Re-work architecture

APPENDIX

- GAN structure: (A)
- Predicting real vs. generated (B)
- Aequitas: (C)
- for privacy: excel sheet

A: GAN Structure

```python
def generator(x,training=True):
  with tf.variable_scope('Generator',reuse=tf.AUTO_REUSE):
    x = tf.layers.dense(x,noise_dim,activation=LeakyReLU)
    x = tf.layers.batch_normalization(x,momentum=0.99)
    x = tf.layers.dense(x,512,activation=LeakyReLU)
    x = tf.layers.batch_normalization(x,momentum=0.99)
    x = tf.layers.dense(x,384,activation=LeakyReLU)
    x = tf.layers.batch_normalization(x,momentum=0.99)
    x = tf.layers.dense(x,384,activation=LeakyReLU)
    x = tf.layers.batch_normalization(x,momentum=0.99)
    out = []
    for i in cat_vector:
      if(i>1):
        out.append(tf.layers.dense(x,i,activation=tf.contrib.sparsemax.sparsemax))
      else:
        out.append(tf.layers.dense(x,1,activation=tf.nn.sigmoid))
    x = tf.layers.flatten(tf.concat(out,1))
    return x

def discriminator(x,training=True):
  with tf.variable_scope('Discriminator',reuse=tf.AUTO_REUSE):
    x = tf.layers.dense(x,noise_dim,activation=LeakyReLU)
    x = tf.layers.dropout(x,0.2)
    x = tf.layers.dense(x,512,activation=LeakyReLU)
    x = tf.layers.dropout(x,0.2)
    x = tf.layers.dense(x,384,activation=LeakyReLU)
    x = tf.layers.dropout(x,0.2)
    x = tf.layers.dense(x,128,activation=LeakyReLU)
    x = tf.layers.dropout(x,0.2)
    x = tf.layers.dense(x,1)
    return x
```

Figure: Structure of GAN (python code)

B1: Chile prediction

- Chile, predicting admissions to university

	real			generated		
GLM	pos.	neg.	Accuracy:	pos.	neg.	Accuracy
pred. pos.	6038.2	11333	0.66672063	5406.2	10398.2	0.73298708
pred. neg.	6069.6	28775.4		3544.2	32867.6	
lasso						
pred. pos.	5431.8	11939.4	0.67218627	4834.2	10970.2	0.73252362
pred. neg.	5177.8	29667.2		2996.4	33415.4	
xgboost						
pred. pos.	5569	11802.2	0.67637655	4243.2	11561.2	0.73163501
pred. neg.	5096.2	29748.8		2451.8	33960	
rf						
pred. pos.	354.4	17016.8	0.66886904	294.6	15509.8	0.70105445
pred. neg.	273.6	34571.4		100	36311.8	

B2: SQF prediction

- SQF, predicting stopped/frisked

	real			generated		
GLM	pos.	neg.	Accuracy:	pos.	neg.	Accuracy
pred. pos.	386.2	342.6	0.74349031	905.2	180.4	0.78075945
pred. neg.	292	1453.2		362	1026.4	
lasso						
pred. pos.	183	545.8	0.73419611	927.6	158	0.77550516
pred. neg.	111.8	1633.4		397.4	991	
xgboost						
pred. pos.	346.6	382.2	0.74680449	913.4	172.2	0.84114782
pred. neg.	244.2	1501		220.8	1167.6	
rf						
pred. pos.	1	727.8	0.70573984	847	238.6	0.81446946
pred. neg.	0.2	1745		220.4	1168	

B3: COMPAS prediction

- COMPAS, predicting recidivism

	real			generated		
GLM	pos.	neg.	Accuracy:	pos.	neg.	Accuracy
pred. pos.	706.4	94.2	0.78369734	236.6	135.6	0.82420628
pred. neg.	172.8	261		81.4	780.8	
lasso						
pred. pos.	721.4	79.2	0.77818588	179.8	192.4	0.79018583
pred. neg.	194.6	239.2		66.6	795.6	
xgboost						
pred. pos.	705.6	95	0.78920643	249	123.2	0.83814259
pred. neg.	165.2	268.6		76.6	785.6	
rf						
pred. pos.	724.2	76.4	0.77510686	210.6	161.6	0.8182166
pred. neg.	201.2	232.6		62.8	799.4	

B4: Schufa prediction

- Schufa, predicting credit worthiness

	real			generated		
GLM	pos.	neg.	Accuracy:	pos.	neg.	Accuracy
pred. pos.	121.8	18.2	0.759	128.6	15	0.7710053
pred. neg.	30	30		30.8	25.6	
lasso						
pred. pos.	123.6	16.4	0.751	132.6	11	0.77401033
pred. neg.	33.4	26.6		34.2	22.2	
xgboost						
pred. pos.	125	15	0.756	131	12.6	0.7590351
pred. neg.	33.8	26.2		35.6	20.8	
rf						
pred. pos.	136.2	3.8	0.713	142	1.6	0.72798455
pred. neg.	53.6	6.4		52.8	3.6	

C1: SQF Aequitas

SQF

	disp._impact	dem._parity	fpr_parity	fnr_parity	ppv_parity	npv_parity	accuracy_parity
sex	0.00+-0.00	-0.50+-0.20	-0.36+-0.01	1.23+-0.05	0.00+-0.00	0.00+-0.00	0.00+-0.00
	-0.07+-0.09	-0.36+-0.21	-0.46+-0.30	2.00+-0.95	-0.03+-0.10	0.68+-0.26	0.03+-0.05
	0.72+-0.28	-0.79+-0.26	nan+-0.52	4.08+-6.37	nan+-0.35	0.46+-0.72	nan+-0.57
age	0.00+-0.00	-0.46+-0.19	-0.32+-0.01	1.15+-0.06	0.00+-0.00	0.00+-0.00	0.00+-0.00
	0.06+-0.07	-0.47+-0.19	-0.26+-0.15	1.45+-0.46	0.01+-0.07	0.22+-0.24	-0.07+-0.05
	0.08+-0.35	-0.43+-0.15	-0.19+-0.71	1.95+-2.24	-0.01+-0.24	0.72+-1.01	-0.18+-0.29
ethnicity	0.00+-0.00	-0.32+-0.24	-0.60+-0.06	0.77+-0.21	0.00+-0.00	0.00+-0.00	0.00+-0.00
	0.10+-0.21	-0.43+-0.21	0.46+-1.49	0.02+-0.30	0.13+-0.11	-0.37+-0.06	0.13+-0.04
	nan+-0.13	nan+-0.19	nan+-0.29	nan+-0.97	nan+-0.16	nan+-0.18	nan+-0.20
	0.00+-0.18	-0.39+-0.21	0.06+-1.35	0.14+-0.36	0.10+-0.10	-0.26+-0.12	0.15+-0.09
	-0.52+-0.16	-0.21+-0.26	-0.24+-0.55	0.27+-0.54	-0.09+-0.10	-0.14+-0.20	0.23+-0.06

C2: COMPAS aequitas

COMPAS

	disp._impact	dem._parity	fpr_parity	fnr_parity	ppv_parity	npv_parity	accuracy_parity
sex	0.00+-0.00	0.90+-0.02	1.34+-0.03	-1.92+-0.12	0.00+-0.00	0.00+-0.00	0.00+-0.00
	-0.22+-0.09	0.92+-0.04	1.05+-0.20	-1.70+-0.75	-0.16+-0.15	-0.10+-0.05	0.15+-0.21
age	0.00+-0.00	0.92+-0.02	1.13+-0.05	-2.17+-0.07	0.00+-0.00	0.00+-0.00	0.00+-0.00
	-0.11+-0.06	0.70+-0.02	1.54+-0.82	-1.90+-0.40	-0.16+-0.18	-0.66+-0.11	0.90+-0.70
	-0.91+-0.42	1.19+-0.26	nan+-0.00	0.00+-0.00	nan+-0.44	-0.67+-0.89	nan+-0.60
ethnicity	0.00+-0.00	0.99+-0.01	2.60+-0.31	-1.19+-0.06	0.00+-0.00	0.00+-0.00	0.00+-0.00
	2.08+-0.33	0.95+-0.01	4.73+-1.24	-0.74+-0.22	0.71+-0.23	0.32+-0.09	-0.33+-0.05
	-0.24+-1.69	1.06+-0.40	-0.34+-10.07	-0.98+-1.04	-0.20+-1.58	0.14+-0.62	0.62+-1.02
	-0.13+-0.19	0.94+-0.05	2.11+-1.78	-1.04+-0.45	-0.11+-0.17	-0.08+-0.06	0.12+-0.41
	-0.13+-0.17	0.75+-0.03	0.95+-1.65	-1.36+-0.55	-0.02+-0.43	-0.22+-0.11	0.31+-0.67

C3: Schufa aequitas

Schufa

	disp._impact	dem._parity	fpr_parity	fnr_parity	ppv_parity	npv_parity	accuracy_parity
sex	0+-0	-0.03+-0.03	-0.11+-0.45	0.01+-0.06	0+-0	0+-0	0+-0
	0.26+-0.26	-0.1+-0.08	-0.5+-2.36	-0.09+-0.35	0.28+-0.36	-0.02+-0.07	-0.06+-0.36
age	0+-0	-0.11+-0.07	-0.26+-0.12	0.05+-0.03	0+-0	0+-0	0+-0
	-0.21+-0.31	-0.04+-0.1	-0.19+-1.56	0.05+-0.16	-0.16+-0.4	0.05+-0.05	0.26+-0.23
	-0.03+-0.87	-0.09+-0.17	3.25+-7.41	0.29+-0.45	-0.52+-1.52	0.03+-0.22	-0.79+-1.7