Operations on Word Vectors

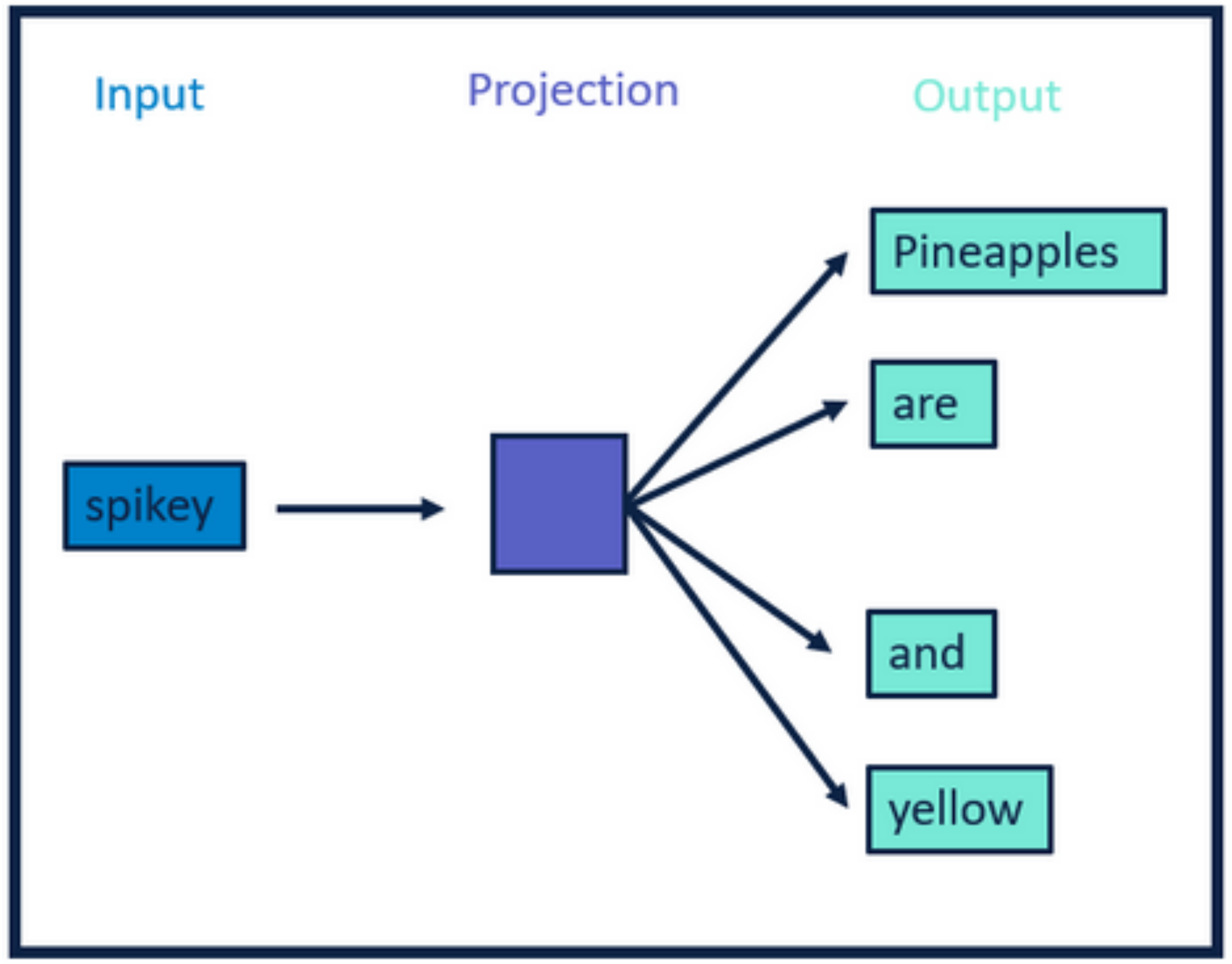
Nitish Joshi, 31st January 2022

Logistics

- **Sections**: 40-50 mins at the end of some lectures (~5/6). Will cover some topics related to lecture + demo/code.
- **Office Hours:** Thursdays 11am-12pm, 60 5th Ave Room 302.

Recap

- Goal: Map each word to a vector in $\mathbb{R}^d$ such that similar words have similar vectors.
- Skip-gram model: Given a word, predict its neighbouring words within a window.

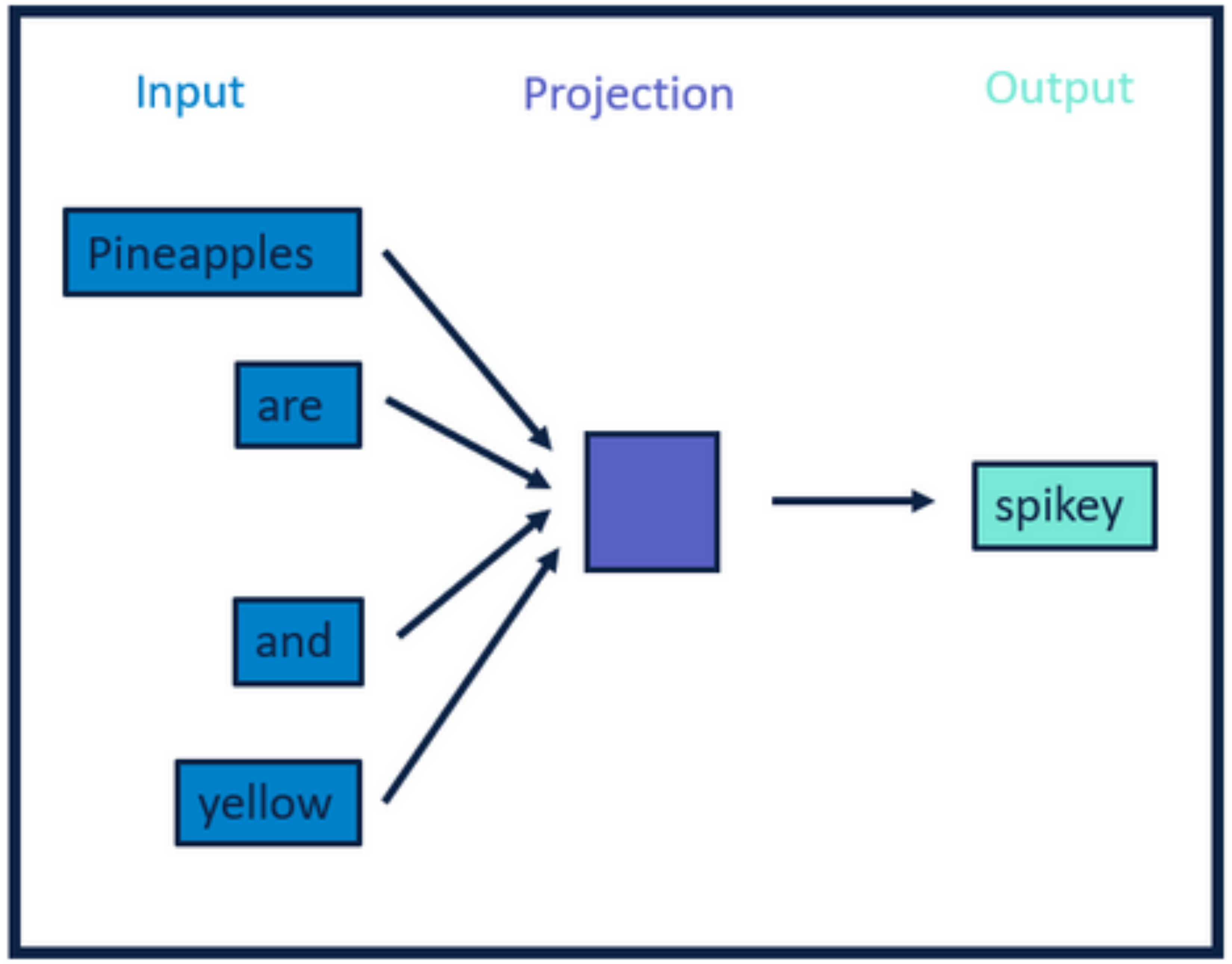


Skip-gram

Source: https://community.alteryx.com/t5/Data-Science/Word2vec-for-the-Alteryx-Community/ba-p/305285

Recap

- Goal: Map each word to a vector in $\mathbb{R}^d$ such that similar words have similar vectors
- Skip-gram model: Given a word, predict its neighbouring words within a window
- Continuous bag-of-words model: Given the context, predict the missing word.



CBOW

Source: https://community.alteryx.com/t5/Data-Science/Word2vec-for-the-Alteryx-Community/ba-p/305285

Recap

- GloVe: Global Vectors (Pennington et al., 2014) — Use co-occurence matrix of each word pair.
- X_{ij} : No. of times the word i occurs in context of j ; w_i : word embedding for i ; c_j : context embedding for j ; $b's$: bias terms.

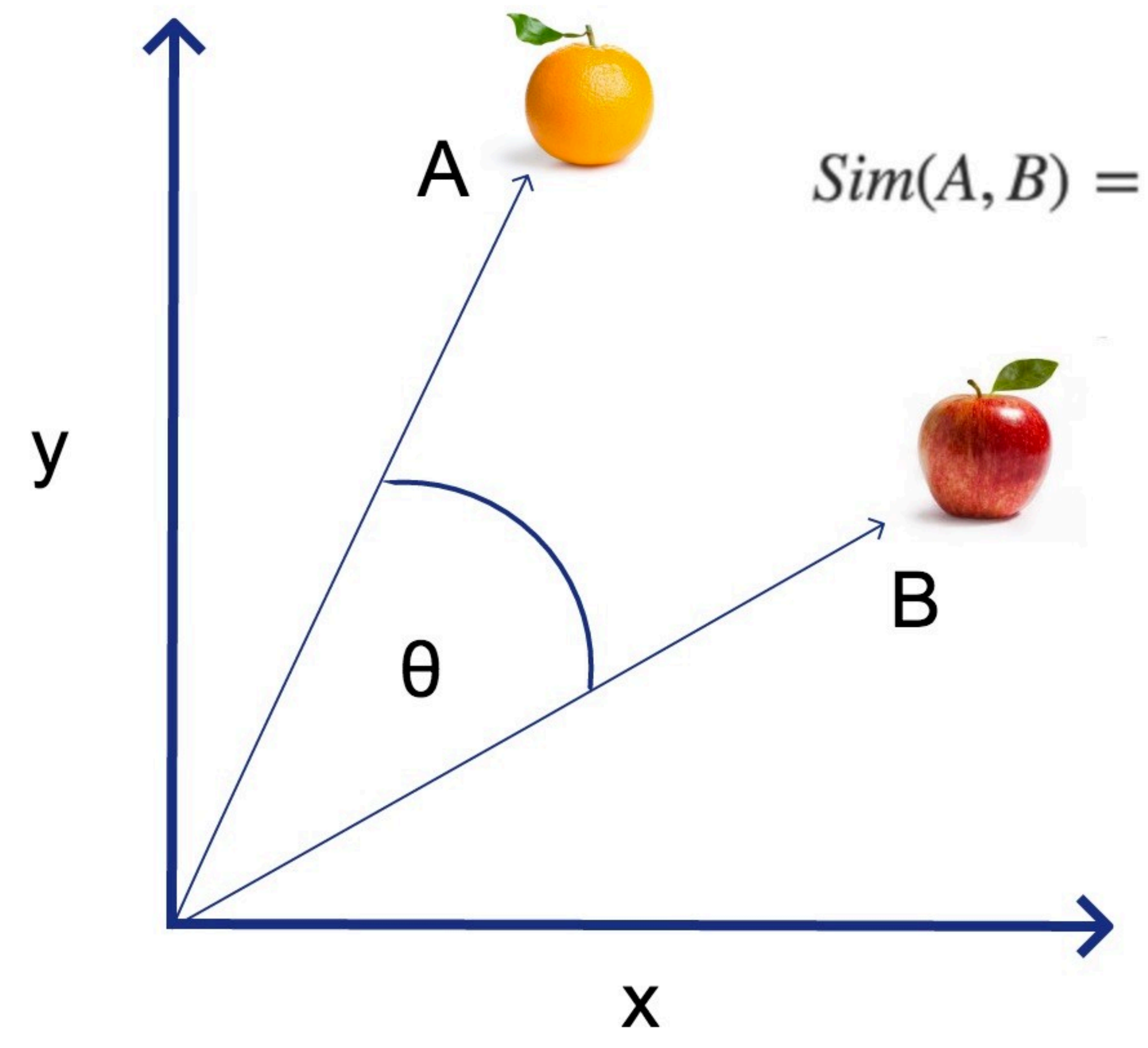
Objective: $J = \sum_{i,j=1}^{V} (w_i^T c_j + b_i + b_j - logX_{ij})^2$

Courtesy: Greg Durrett (UT Austin)

Similarity between word vectors

- **Question**: Do the learnt word embeddings satisfy the desired property of similarity?
- Use cosine similarity between any two word vectors.

Cosine Similarity



$$Sim(A, B) = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|}$$

Source: https://www.linkedin.com/pulse/cosine-similarity-classification-michael-lin/

Word Analogy Task

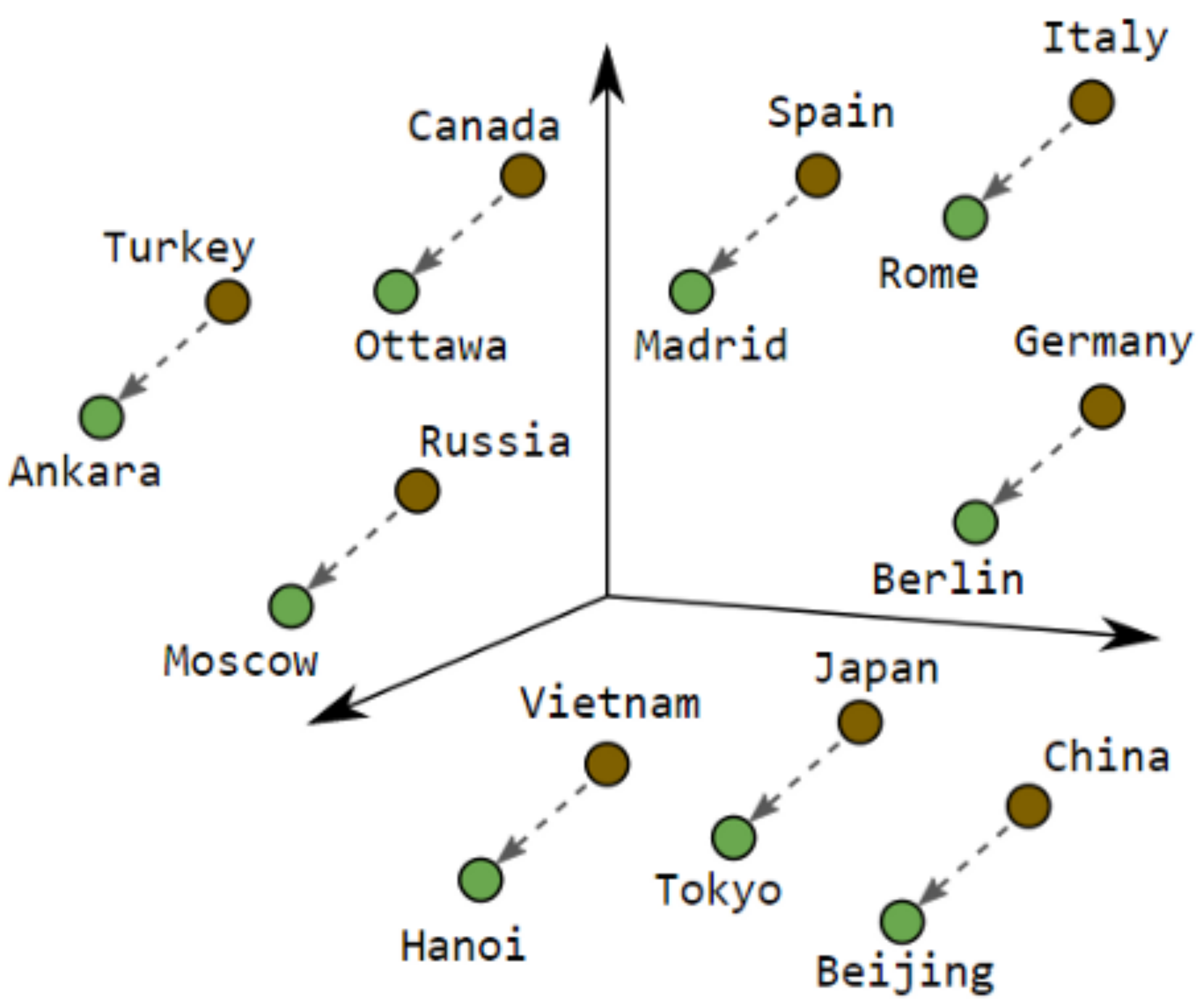
- In word analogy tasks, we ask questions like “a is to b as c is to ___”
- Example: “London is to UK as Amsterdam is to <u>Netherlands</u>”

Word Analogy Task

- For a —> b :: c —> ?, given word vectors v_a, v_b and v_c , we will find a word d such that v_a - v_b ~ v_c - v_d
- The difference v_a - v_b represents the ‘concept’ (e.g. capital of country).

Word Analogy Task

- For a —> b :: c —> ?, given word vectors v_a, v_b and v_c , we will find a word d such that v_a - v_b ~ v_c - v_d
- The difference v_a - v_b represents the 'concept' (e.g. capital of country).



Country-Capital

Source: https://developers.google.com/machine-learning/crash-course/images/linear-relationships.svg

Bias in word vectors

- The difference v_a - v_b represents the ‘concept’ — if a is woman and b is man, then it represents ‘gender’.
- Compute projections of occupations on this difference v_a - v_b .

Extreme *she* occupations

1. homemaker	2. nurse	3. receptionist
4. librarian	5. socialite	6. hairdresser
7. nanny	8. bookkeeper	9. stylist
10. housekeeper	11. interior designer	12. guidance counselor

Extreme *he* occupations

1. maestro	2. skipper	3. protege
4. philosopher	5. captain	6. architect
7. financier	8. warrior	9. broadcaster
10. magician	11. figher pilot	12. boss

Bolukbasi et al. 2016

Bias in word vectors

- Similarly, we can obtain vectors for other concepts like race and religion.
- Compute projections of occupations on this difference v_a - v_b .

Racially Biased Analogies	
black → criminal	caucasian → police
asian → doctor	caucasian → dad
caucasian → leader	black → led
Religiously Biased Analogies	
muslim → terrorist	christian → civilians
jewish → philanthropist	christian → stooge
christian → unemployed	jewish → pensioners

Manzini et al., 2019

Note: The vectors were obtained from training on reddit data from USA users

Debiasing Word Vectors

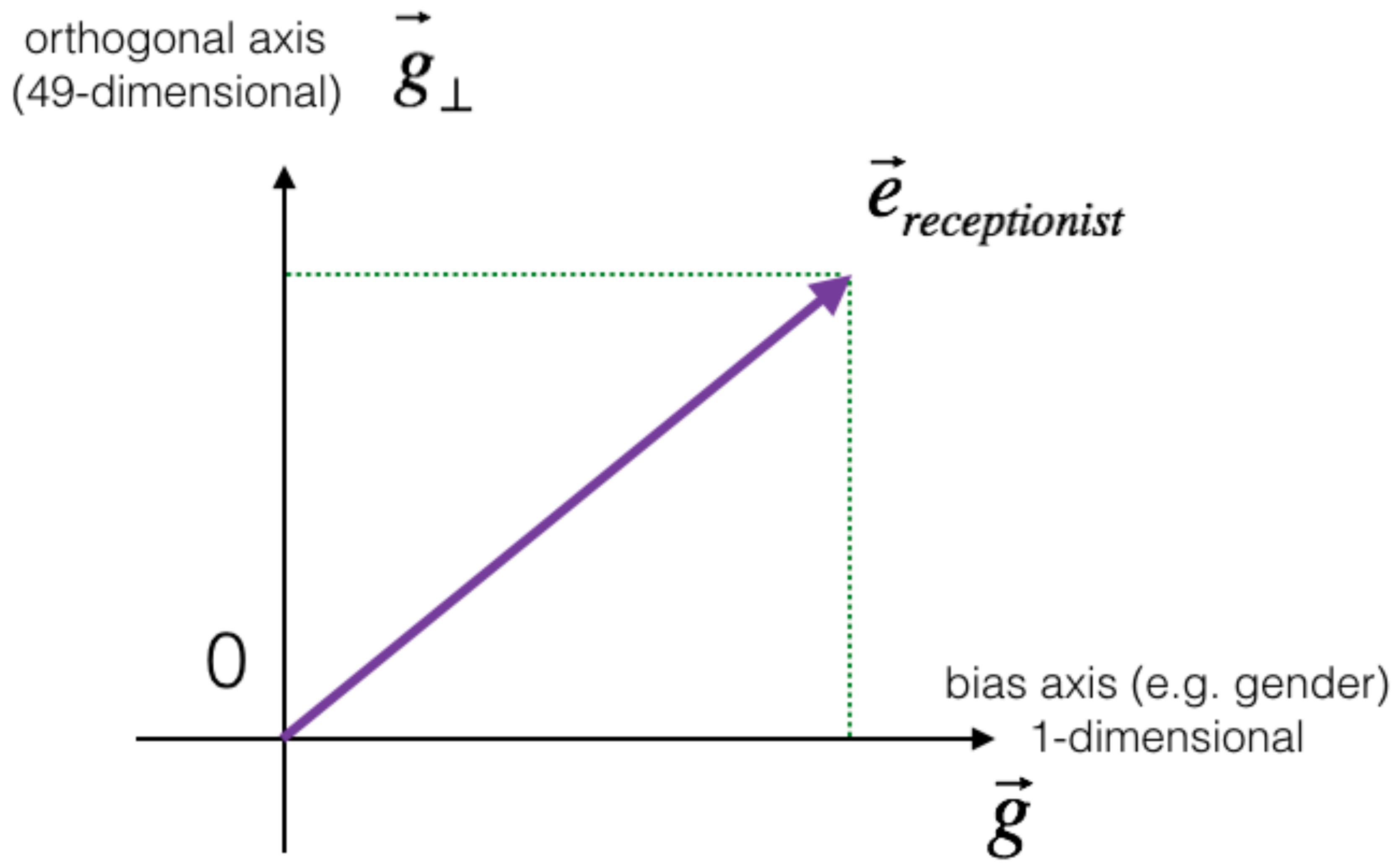
- For a concept vector g and word vector e, obtain the biased component:

$$e_{\text{biased}} = \frac{e \, . \, g}{||g||} g$$

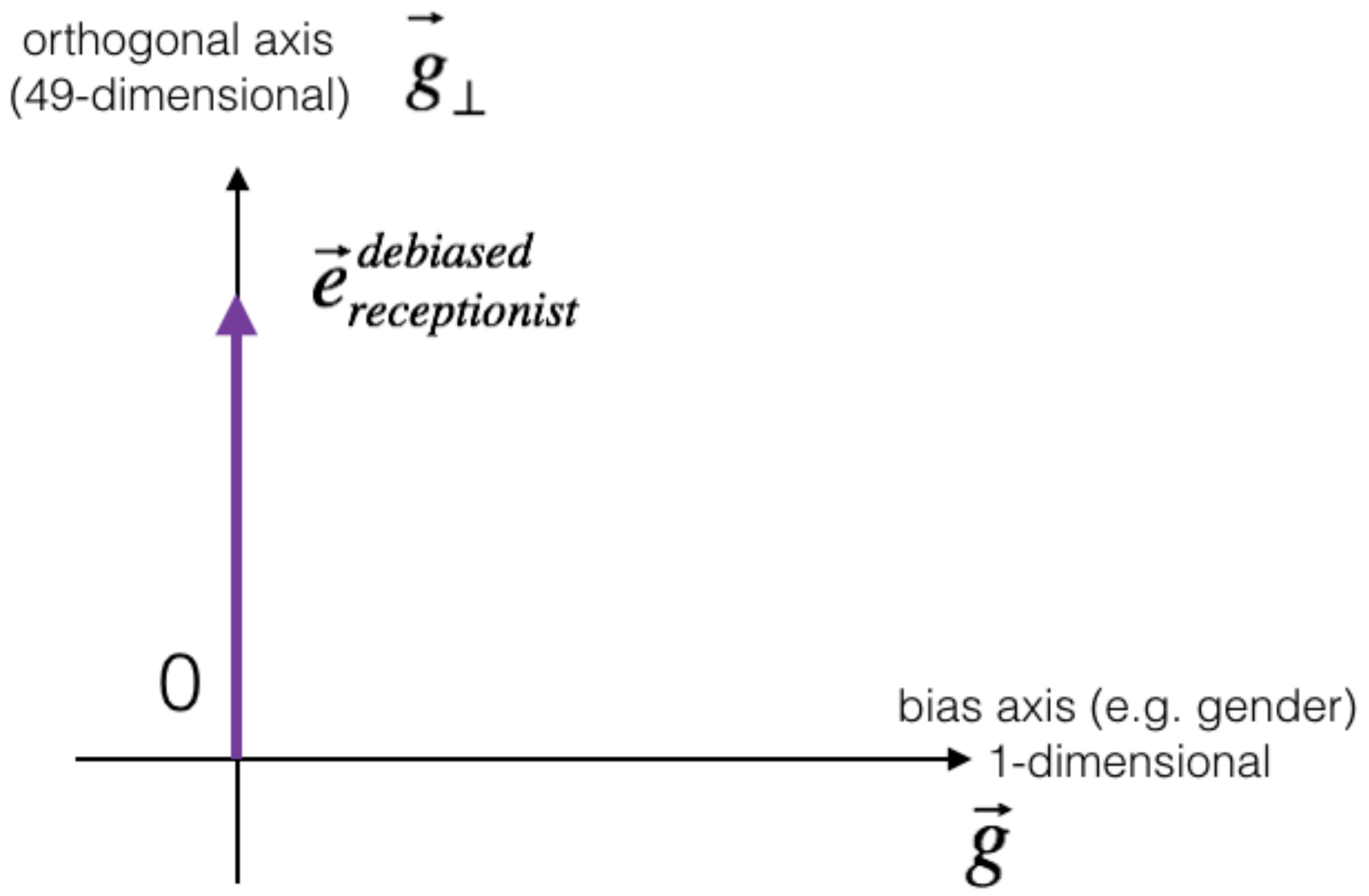
- Subtract from the original vector to obtain the debiased vector

$$e_{\text{debiased}} = e - e_{\text{biased}}$$

Debiasing Word Vectors



before neutralizing,
"receptionist" is positively correlated with the bias axis

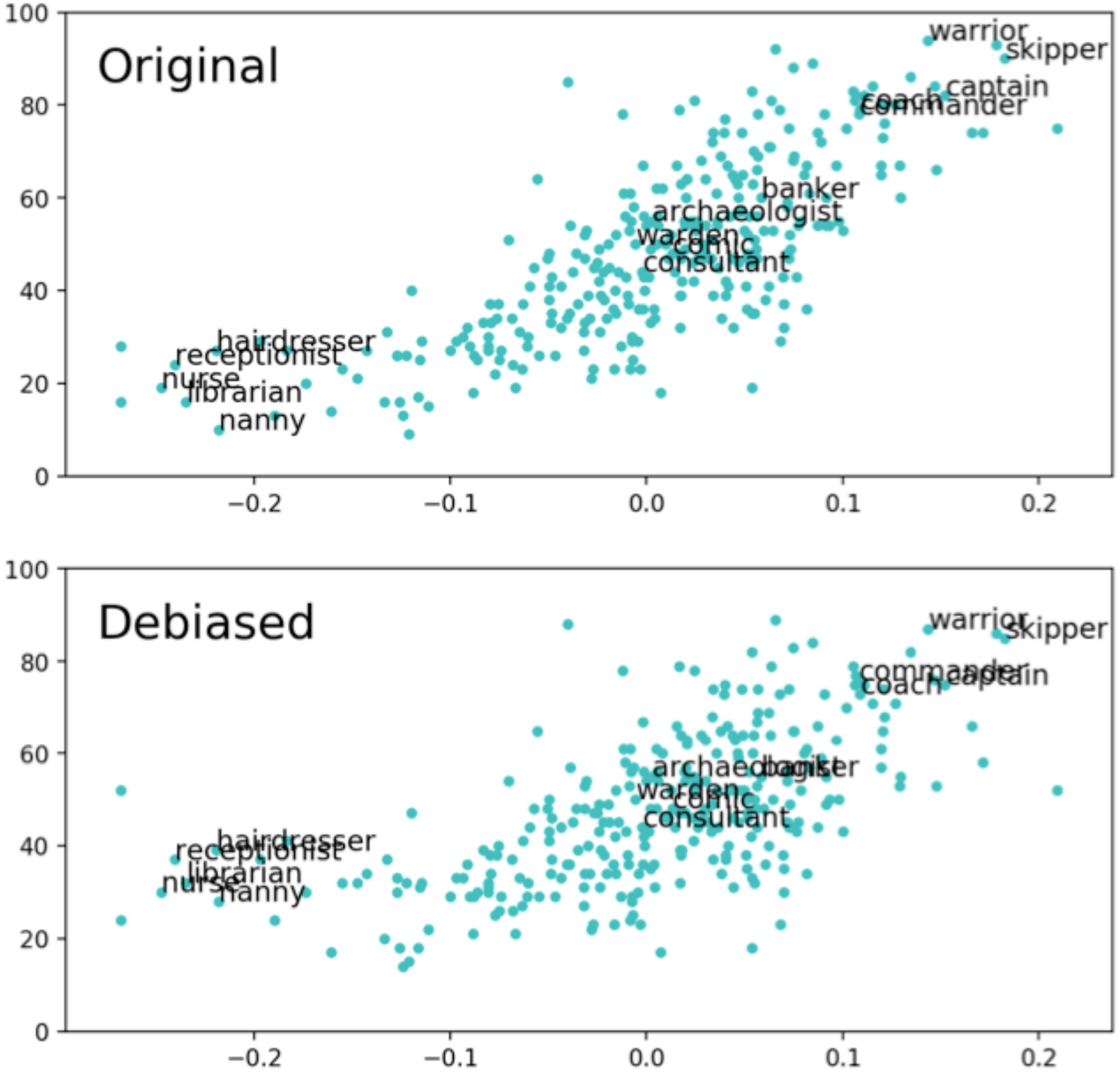


after neutralizing,
debased version, with the component
in the direction of the bias axis (g) zeroed out

Source: https://zhangruochi.com/Operations-on-word-vectors-Debiasing/2019/03/28/

Debiasing Word Vectors

- Previous method ensures that vector is orthogonal to the concept vector.
- Not always effective in debiasing —- the word vectors corresponding to occupations are still clustered according to gender.



Gonen and Goldberg, 2019

Other Debiasing Methods

- **Ravfogel et al 2020**:
  - There is no single direction corresponding to concepts — it can span in multiple directions.
  - Propose Iterative Null-space Projection (INLP) — iteratively neutralise/ debias the vectors.

Other Debiasing Methods: INLP

Algorithm 1 Iterative Nullspace Projection (INLP)

Input : (X, Z): a training set of vectors and protected attributes
n: Number of rounds

Result: A projection matrix P

Function `GetProjectionMatrix(`X, Z`)`**:**

$X_{projected} \leftarrow X$
$P \leftarrow I$
for $i \leftarrow 1$ **to** n **do**
$W_i \leftarrow$ TrainClassifier($X_{projected}, Z$)
$B_i \leftarrow$ GetNullSpaceBasis(W_i)
$P_{N(W_i)} \leftarrow B_i Bi^T$
$P \leftarrow P_{N(W_i)} P$
$X_{projected} \leftarrow P_{N(W_i)} X_{projected}$
end
return P

e.g. Dataset of (occupation, gender) where we have word vectors for each occupation along with the biased gender.

Ravfogel et al., 2020

Other Debiasing Methods: INLP

Algorithm 1 Iterative Nullspace Projection (INLP)

Input : (X, Z): a training set of vectors and protected attributes
n: Number of rounds

Result: A projection matrix P

Function `GetProjectionMatrix(`X, Z`)`**:**

$X_{projected} \leftarrow X$

$P \leftarrow I$

for $i \leftarrow 1$ **to** n **do**

$W_i \leftarrow \text{TrainClassifier}(X_{projected}, Z)$

$B_i \leftarrow \text{GetNullSpaceBasis}(W_i)$

$P_{N(W_i)} \leftarrow B_i Bi^T$

$P \leftarrow P_{N(W_i)} P$

$X_{projected} \leftarrow P_{N(W_i)} X_{projected}$

end

return P

e.g. Train a linear classifier to predict gender from occupation.

Ravfogel et al., 2020

Other Debiasing Methods: INLP

Algorithm 1 Iterative Nullspace Projection (INLP)

Input : (X, Z): a training set of vectors and protected attributes
n: Number of rounds

Result: A projection matrix P

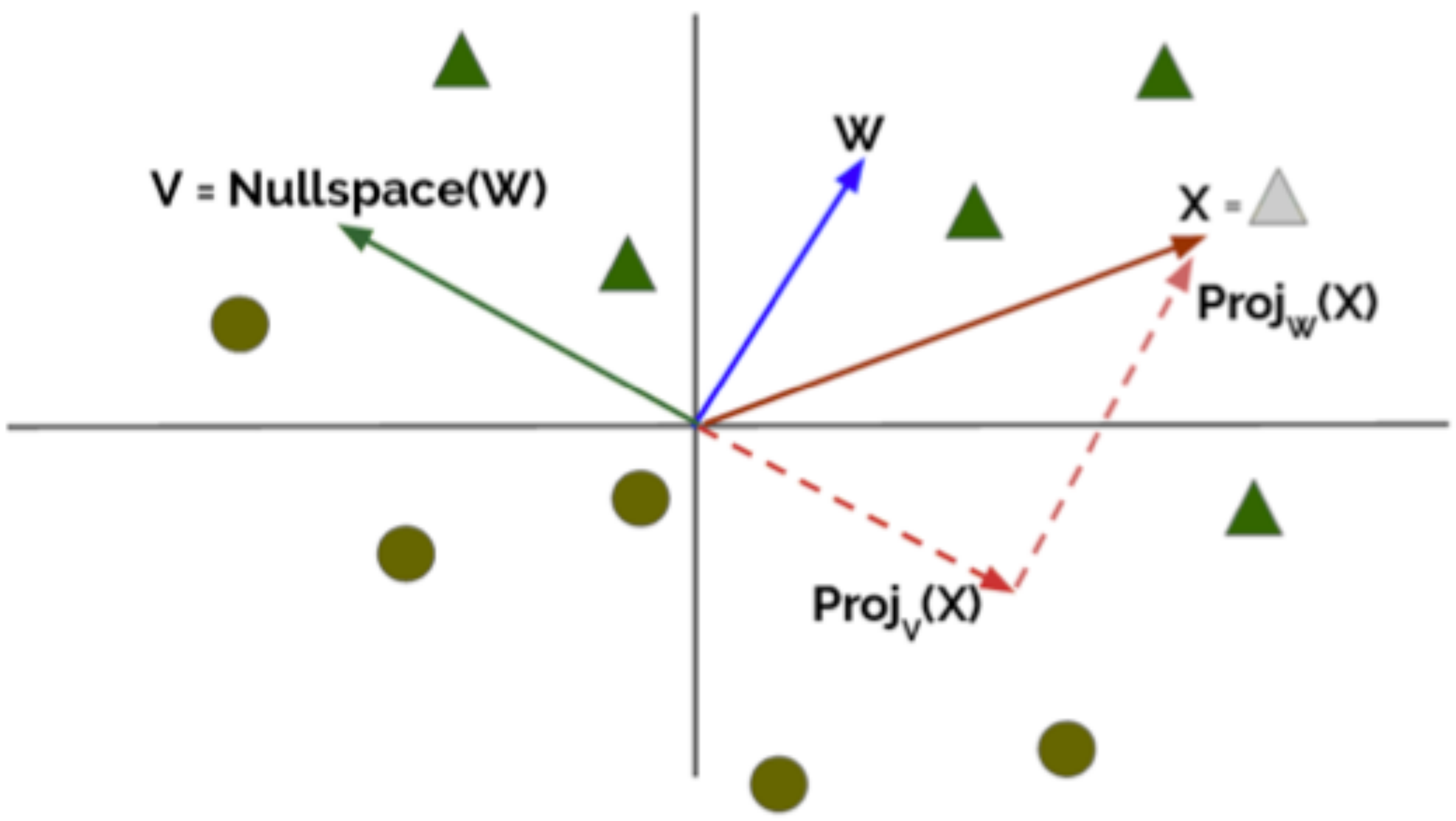
Function `GetProjectionMatrix(`X, Z`)`**:**

$X_{projected} \leftarrow X$
$P \leftarrow I$
for $i \leftarrow 1$ **to** n **do**
$W_i \leftarrow$ TrainClassifier($X_{projected}, Z$)
$B_i \leftarrow$ GetNullSpaceBasis(W_i)
$P_{N(W_i)} \leftarrow B_i Bi^T$
$P \leftarrow P_{N(W_i)} P$
$X_{projected} \leftarrow P_{N(W_i)} X_{projected}$
end
return P

Project X onto nullspace of W —> predicting Z (e.g. gender) from new X will not work.

Ravfogel et al., 2020

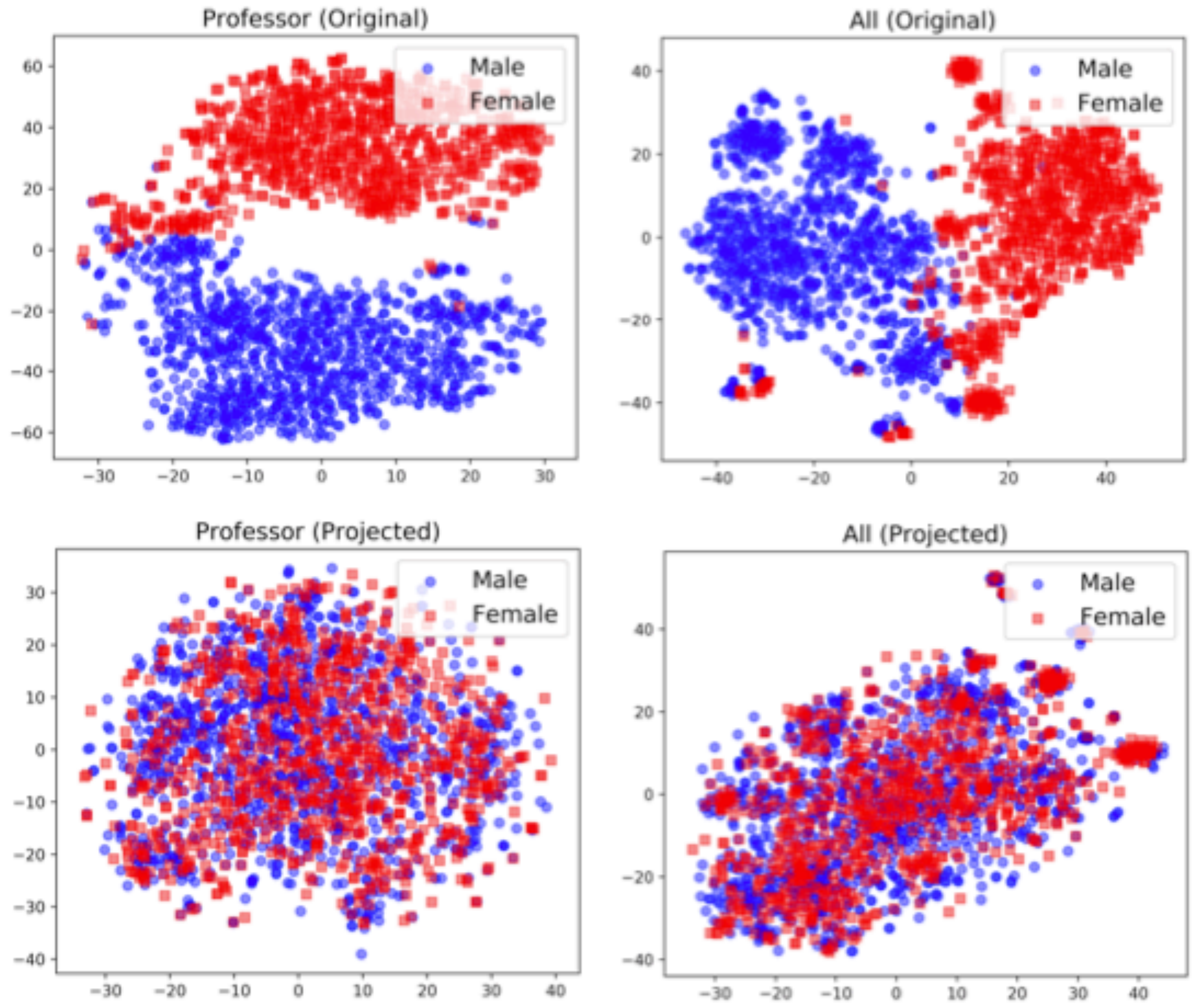
Other Debiasing Methods: INLP



- W: weight of a linear classifier trained to predict Z from X
- Project on null-space
- Iterate

Other Debiasing Methods: INLP

- Does not suffer from the issue we saw with earlier debiasing method.
- Representations are now not clustered according to protected attribute (e.g. gender).



Ravfogel et al., 2020

Summary

- Word vectors encode a notion of similarity, which can be helpful for retrieval, word analogy tasks etc.

- Word vectors can encode biases from the data —> Need to evaluate and use appropriate debiasing methods.