

# TFG informe inicial

Raquel

November 9, 2017

## Contents

<b>1</b>	<b>Data</b>	<b>1</b>
1.1	Initial Data . . . . .	1
1.2	Generated Data . . . . .	1
<b>2</b>	<b>Objetivos</b>	<b>2</b>
<b>3</b>	<b>Análisis Base</b>	<b>3</b>
3.1	Intra sinset . . . . .	3
3.2	Distribución total de las features . . . . .	6
3.3	Distribución de las features por layer . . . . .	6
3.4	Distribución de las features por synset . . . . .	9
3.5	Features per image . . . . .	11
3.6	Images per feature . . . . .	13
3.7	Images per feature per synset . . . . .	17
3.8	Comprobación de que las cosas tienen sentido . . . . .	25
3.9	Estudio de los outliers de imágenes per feature . . . . .	25
3.10	Matrices de cambio . . . . .	25

## 1 Data

### 1.1 Initial Data

- Embedding matrix of size (50000, 12416), con 62080000 features.
- **labels** Labels vector of size  $50k$  which every label is in numeric format (0, 999)
- **synsets = synset0 synset1 synset2 ...** The set of synsets that we will analyze:  
*synsets = [living\\_things, mammal, dog, hunting\\_dogs]*
- **categories = {-1 0 1 }** The possible values of the features.
- **The layers:**

```
self.layers = {
    'conv1_1': [0, 64],  # 1
    'conv1_2': [64, 128], # 2
    'conv2_1': [128, 256], # 3
    'conv2_2': [256, 384], # 4
    'conv3_1': [384, 640], # 5
    'conv3_2': [640, 896], # 6
    'conv3_3': [896, 1152], # 7
    'conv4_1': [1152, 1664], # 8
    'conv4_2': [1664, 2176], # 9
    'conv4_3': [2176, 2688], # 10
    'conv5_1': [2688, 3200], # 11
    'conv5_2': [3200, 3712], # 12
    'conv5_3': [3712, 4224], # 13
```

```

'fc6': [4224, 8320], # 14
'fc7': [8320, 12416], # 15
'conv1': [0, 128], # 16
'conv2': [128, 384], # 17
'conv3': [384, 1152], # 18
'conv4': [1152, 2688], # 19
'conv5': [2688, 4224], # 20
'conv': [0, 4224], # 21
'2_5conv': [128, 4224], # 22
'fc6tofc7': [4224, 12416], # 23
# 'all':[0,12416]           # 24
}

```

## 1.2 Generated Data

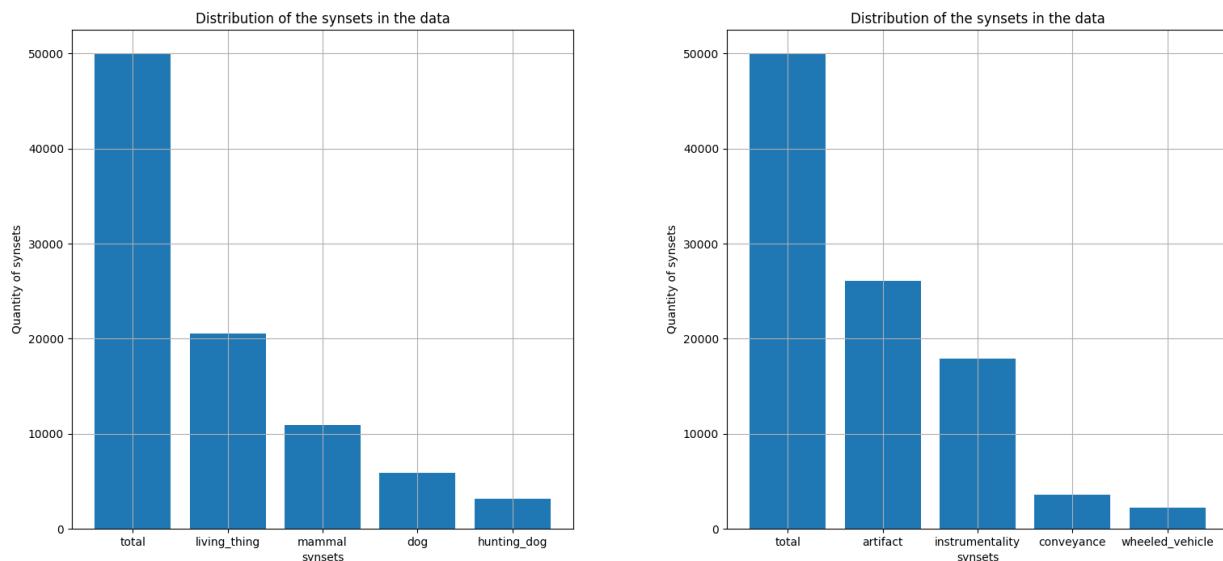
- **synset\_index\_hyponim.txt** A list with all the hyponyms of every synset.
- **synset\_index.txt** For each synset a list with the index of the elements of the hyponym list in the embedding.
- Un diccionario con la cantidad de imágenes que tiene cada feature para cada category.
- **features\_per\_image[synsets].pkl** dfd
- **features\_per\_layer[synsets].pkldsf**
- **Images/\_per\_feature\_per\_synset[synsets].pkl** Genera un diccionario tal que:  
dict[features][category][synset] = cantidad e imagenes del synset que tienen esta categoria en cuestión.
- **intra\_synset[synsets].pklfdsfs**

## 2 Objetivos

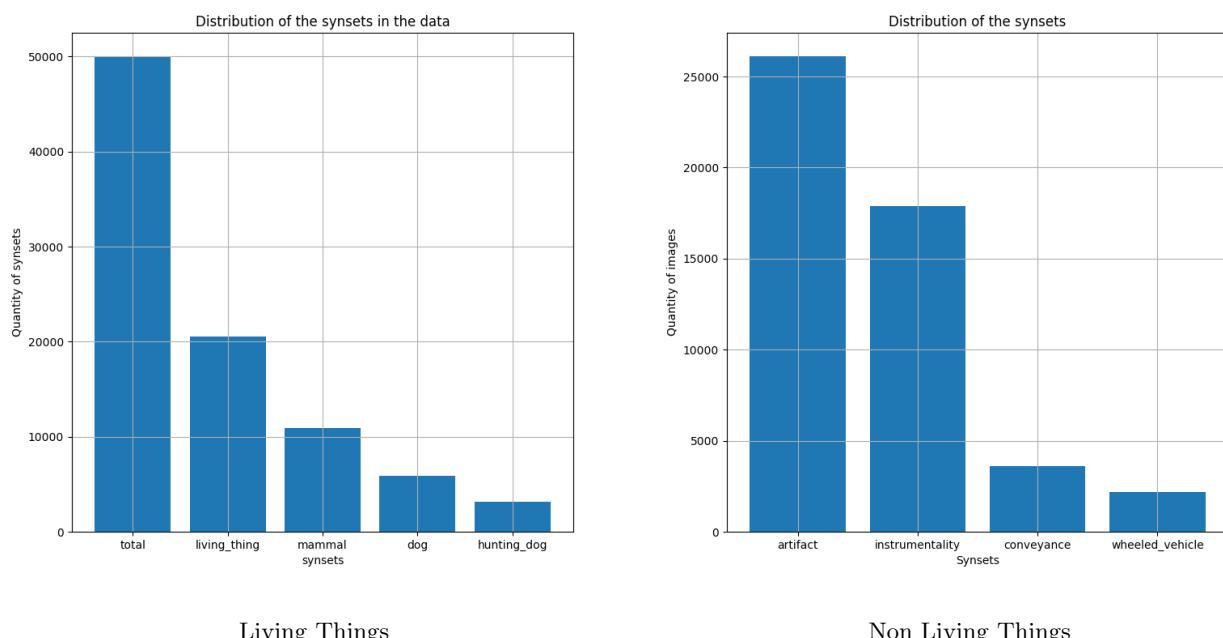
- Estadísticas del dataset por imagen del conjunto de synsets (**hecho**)
- Estadísticas internas para cada synset(que synsets son subconjunto de cual y en que proporción) (**hecho**)
- Estadísticas totales de la distribución de las features respecto a toda la matriz de imágenes. (**solo faltan grafiquillas**)
- Estadísticas totales de la distribución de las features respecto a toda la matriz de imágenes por layer. (**falta escribir + grafiquillas**)
- Estadísticas de las features respecto el subconjunto de imágenes de cada synset. (**faltan grafiquillas**)
- Estadísticas de las features respecto el subconjunto de imágenes de cada synset por layer. (**falta todo**)
- Distribución de imágenes por feature (para 1, -1, 0):
  - respecto a todas las imágenes
  - respecto a cada uno de los subconjuntos de imágenes de cada synset
- **está calculado, falta escribir**
- Repetir para el resto de embeddings
- Montarlo para que me genere todo automático para cada conjunto de synsets.
- Hacer que guarde todo en maps
- Features por imagen: Para cada imagen cuantas features (de cada category) se activan. Usando todos los valores obtenidos dibujo un histograma para cada category de feature. Con el eje de las x siendo la frecuencia y el de las y la cantidad de imágenes que cumplen eso.
- Imágenes por feature: Para cada features (de cada category), cuantas imágenes toman este valor. Usando todos los valores obtenidos dibujo un histograma para cada category de feature. Con el eje de las x siendo la frecuencia y el de las y la cantidad de imágenes que cumplen eso.

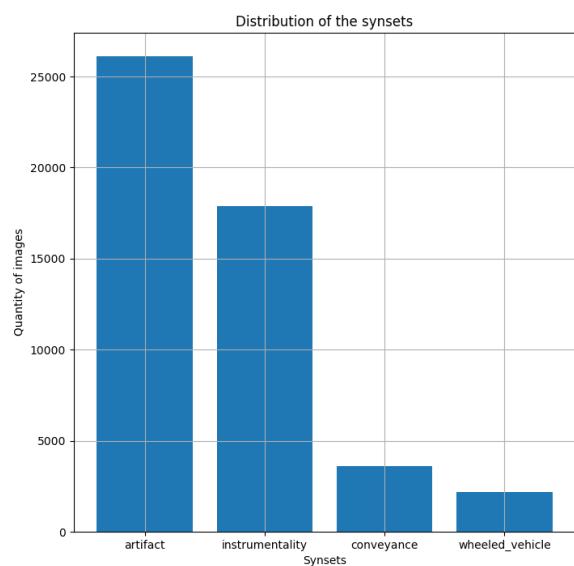
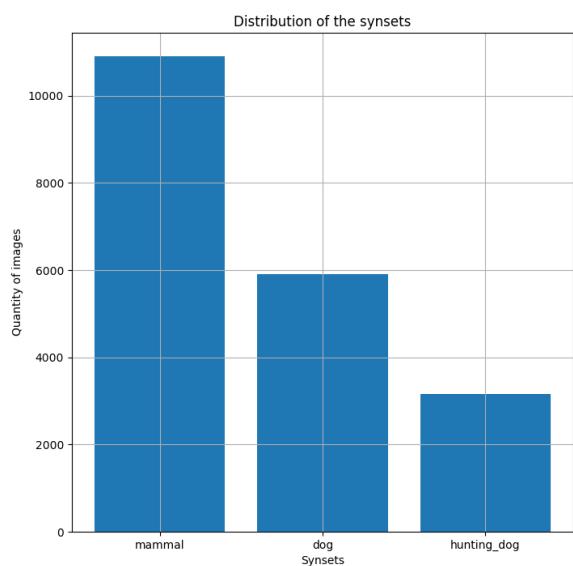
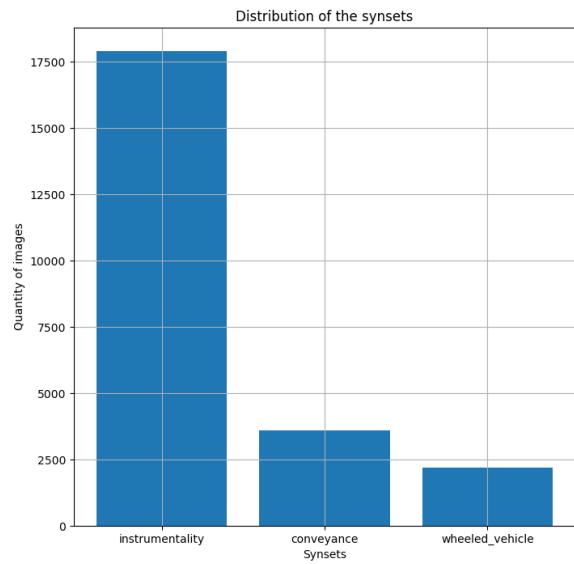
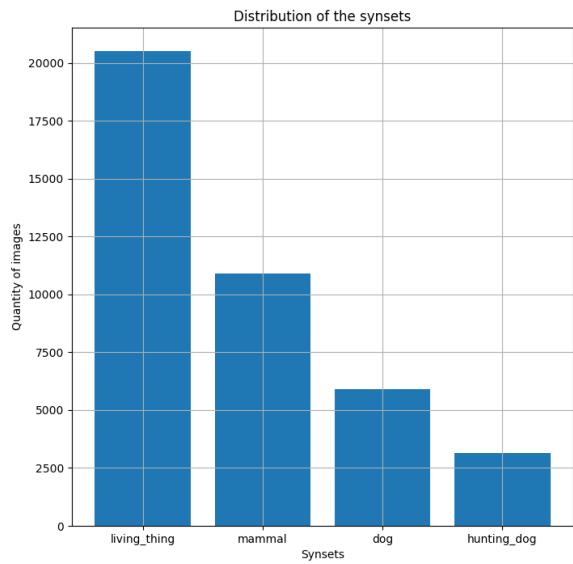
- Comprobar si dentro de un mismo synset hay features que se den con 1 y -1.
- Hacer las distribuciones de la suma.
- Ordenar del código y generar una documentación y tests.

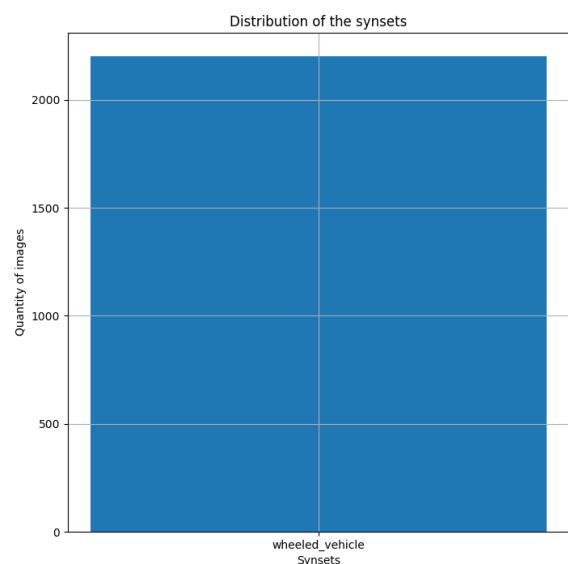
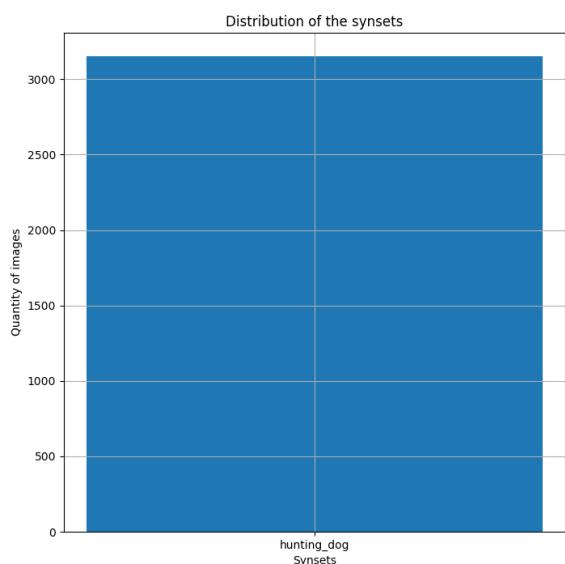
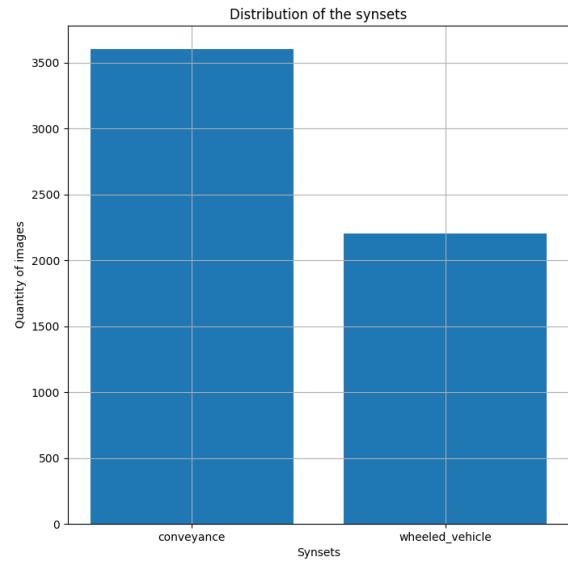
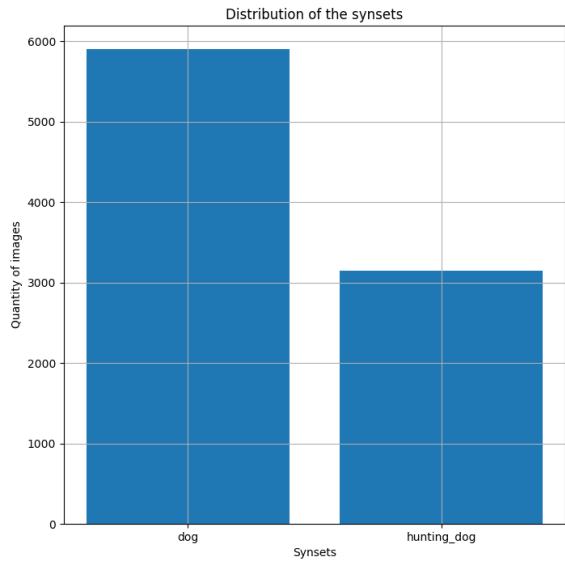
### 3 Análisis Base



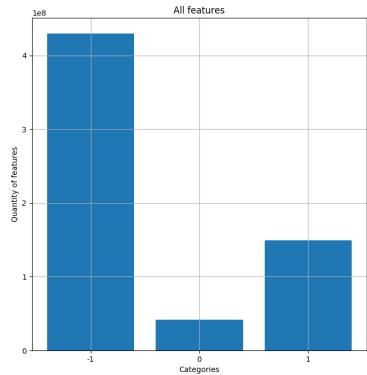
#### 3.1 Intra sinset



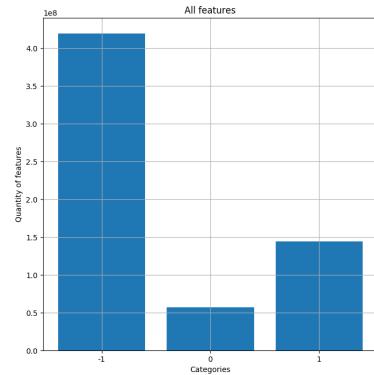




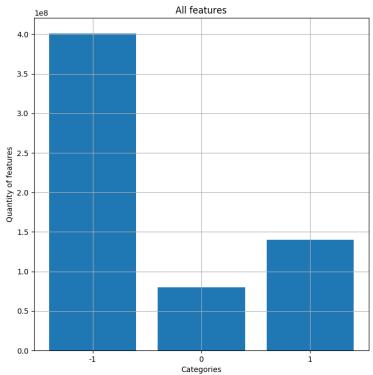
## 3.2 Distribución total de las features



Embedding 19

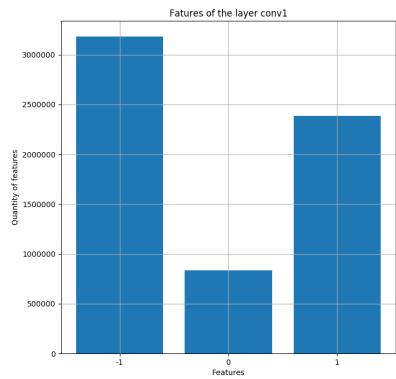


Embedding 25

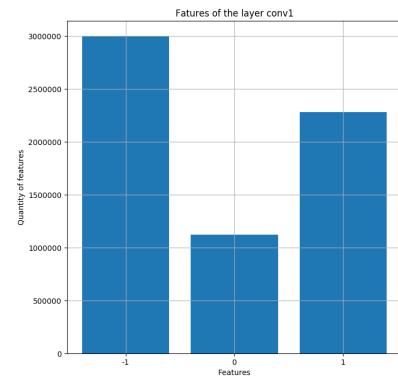


Embedding 31

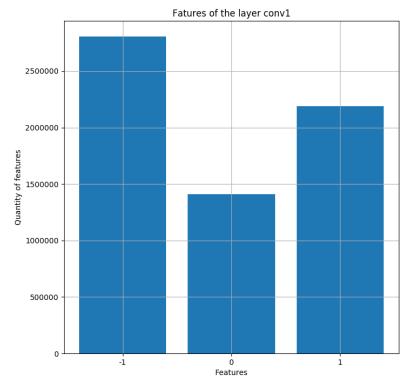
## 3.3 Distribución de las features por layer



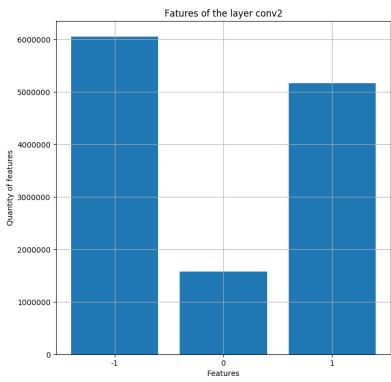
Embedding 19



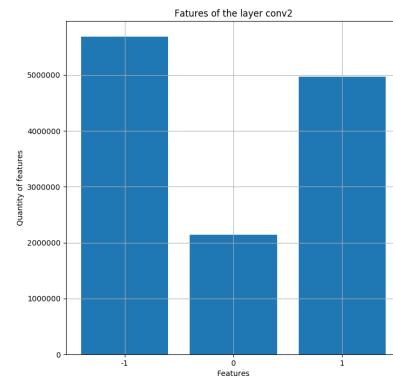
Embedding 25



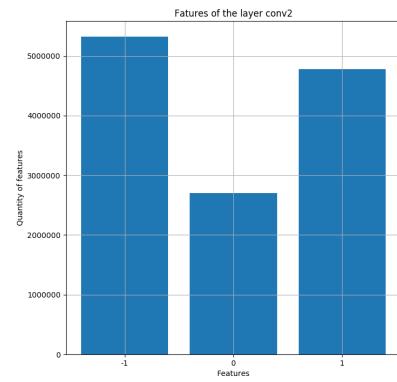
Embedding 31



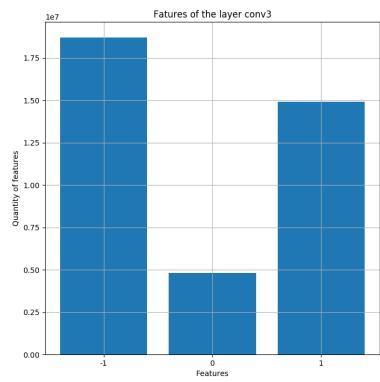
Embedding 19



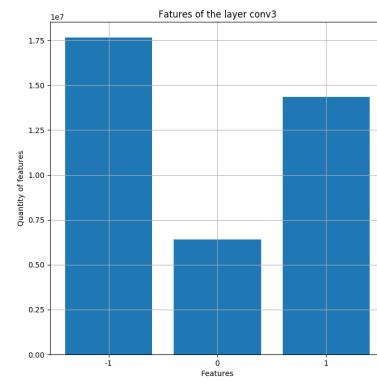
Embedding 25



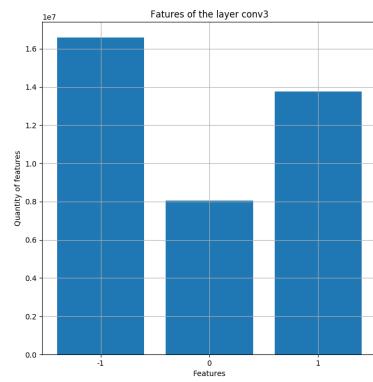
Embedding 31



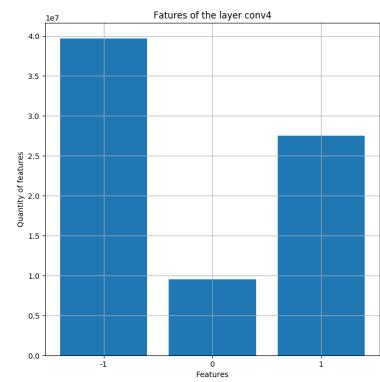
Embedding 19



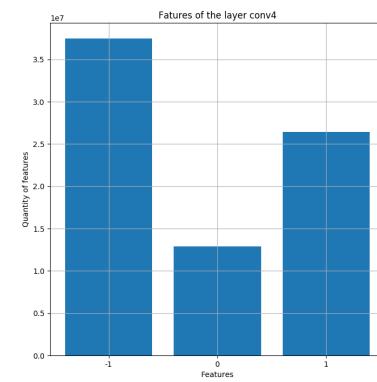
Embedding 25



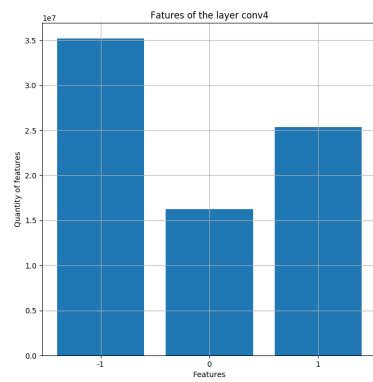
Embedding 31



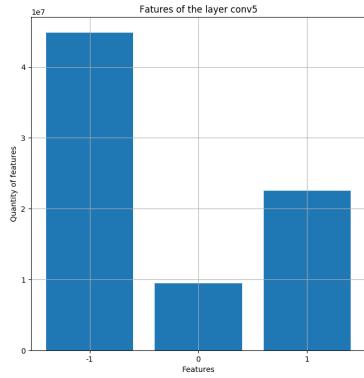
Embedding 19



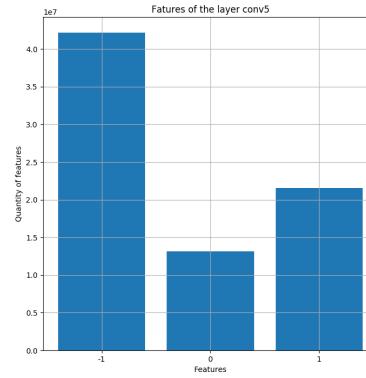
Embedding 25



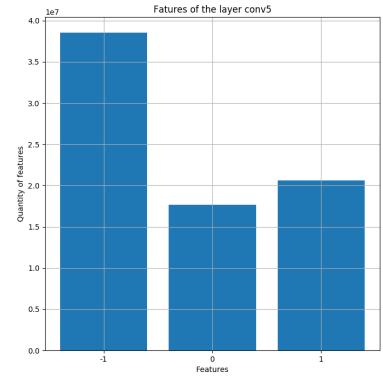
Embedding 31



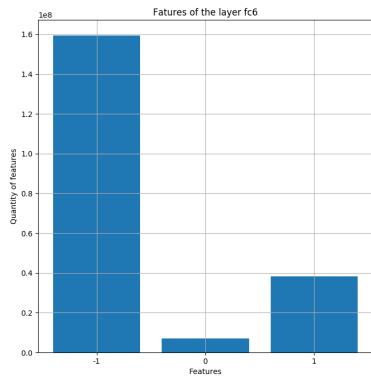
Embedding 19



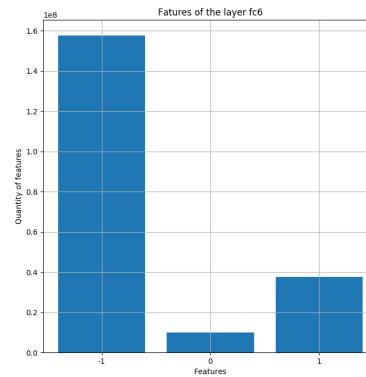
Embedding 25



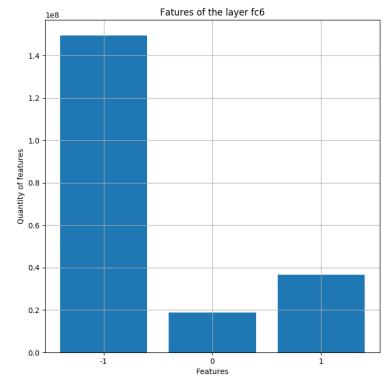
Embedding 31



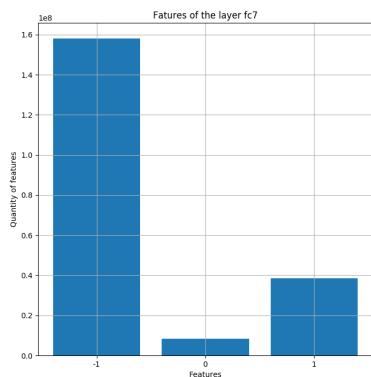
Embedding 19



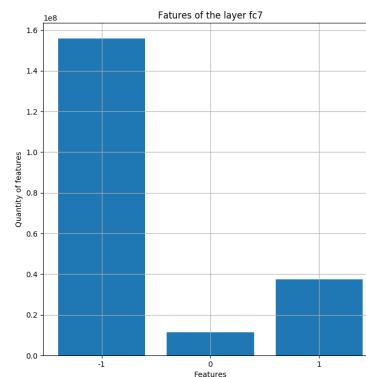
Embedding 25



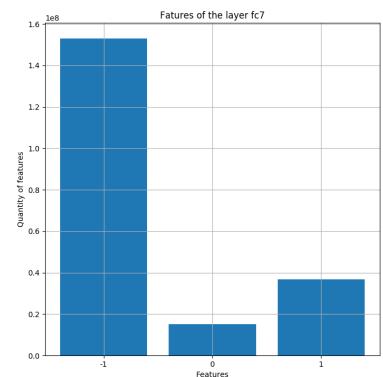
Embedding 31



Embedding 19

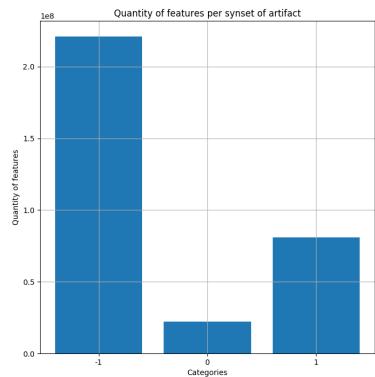


Embedding 25

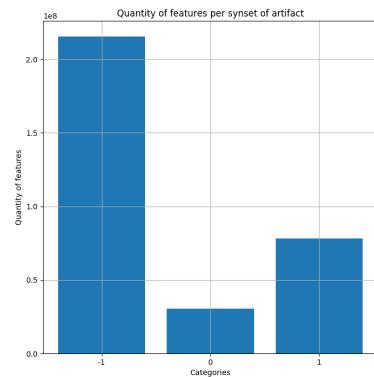


Embedding 31

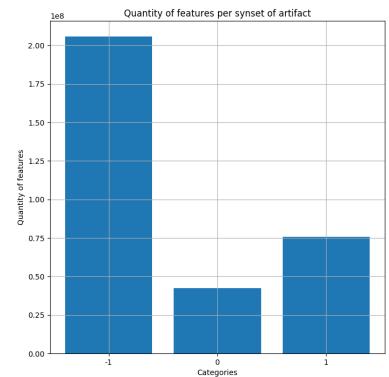
### 3.4 Distribución de las features por synset



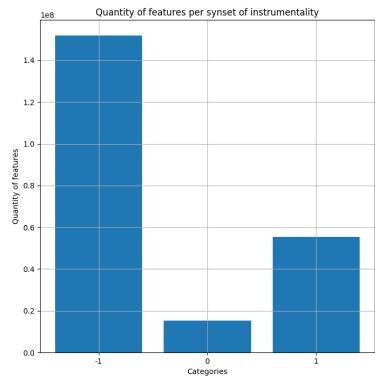
Embedding 19



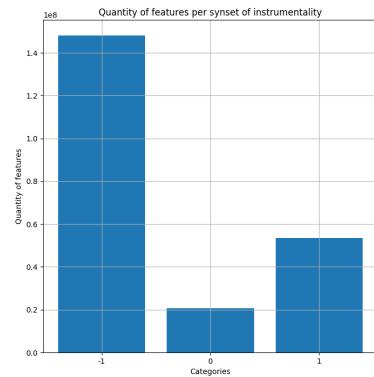
Embedding 25



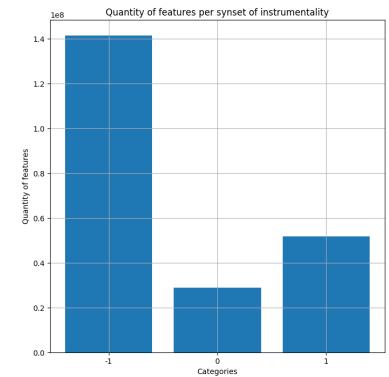
Embedding 31



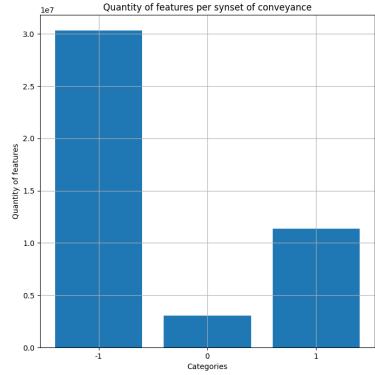
Embedding 19



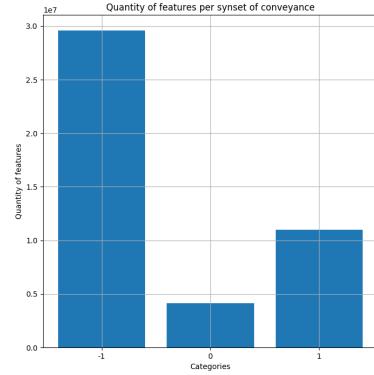
Embedding 25



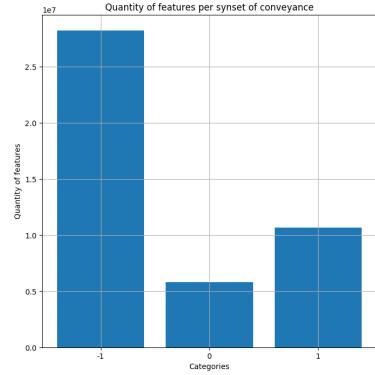
Embedding 31



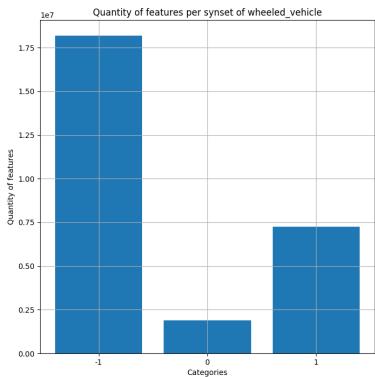
Embedding 19



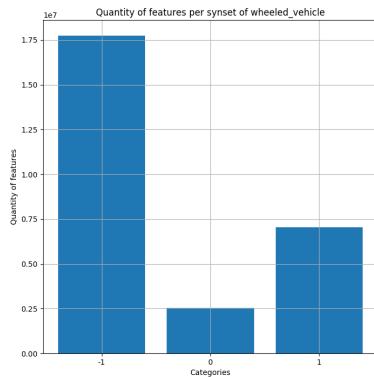
Embedding 25



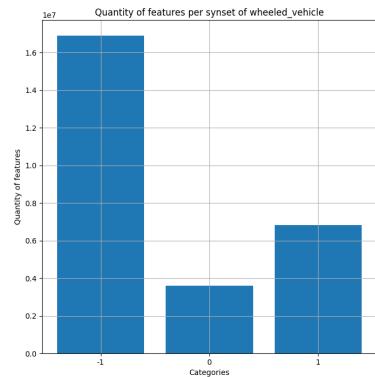
Embedding 31



Embedding 19



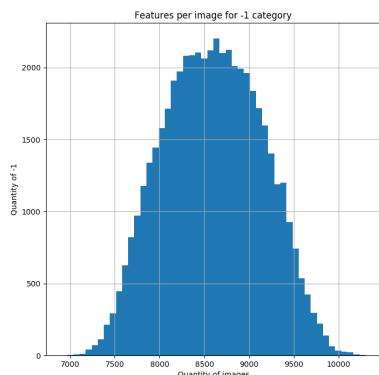
Embedding 25



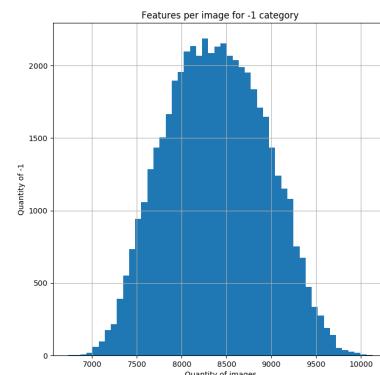
Embedding 31

### 3.5 Features per image

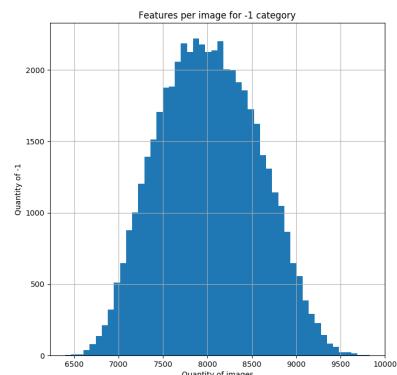
There are for each image the total counting of features active, for the different possible categories ( $-1, 0, 1$ ):



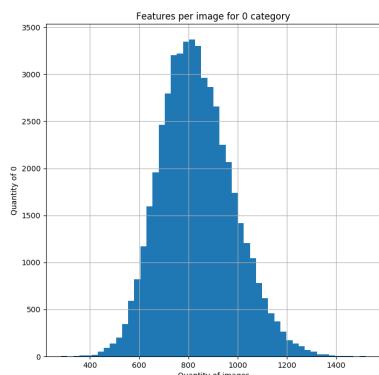
Embedding 19



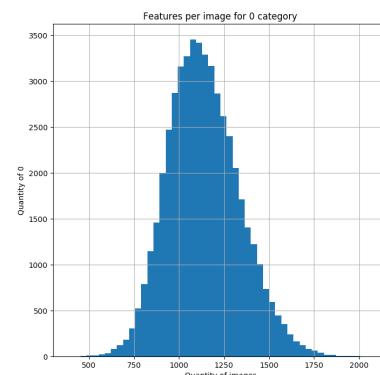
Embedding 25



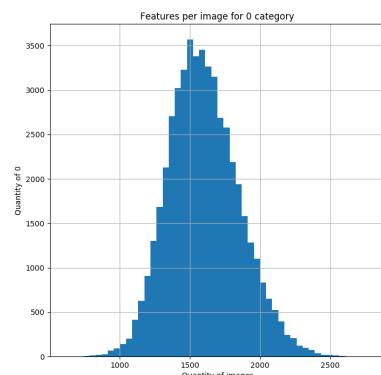
Embedding 31



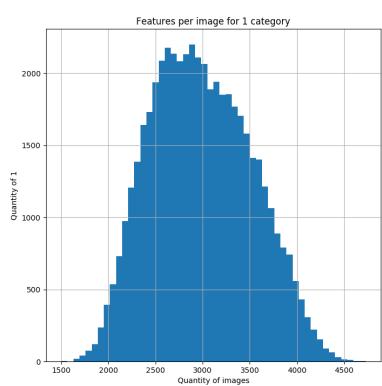
Embedding 19



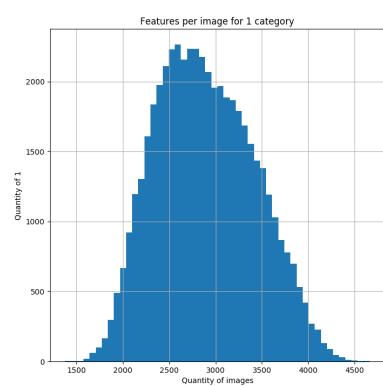
Embedding 25



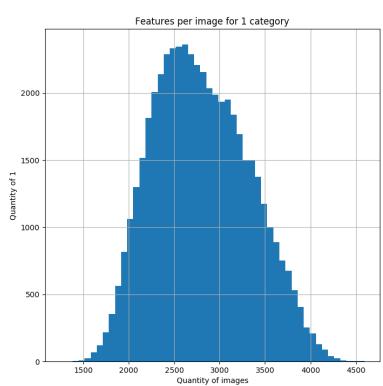
Embedding 31



Embedding 19



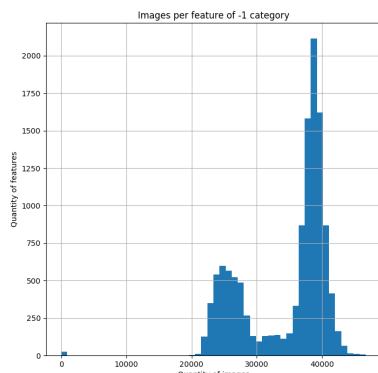
Embedding 25



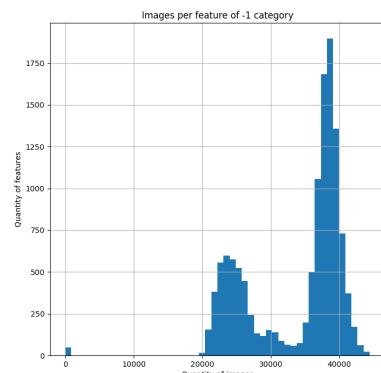
Embedding 31

### 3.6 Images per feature

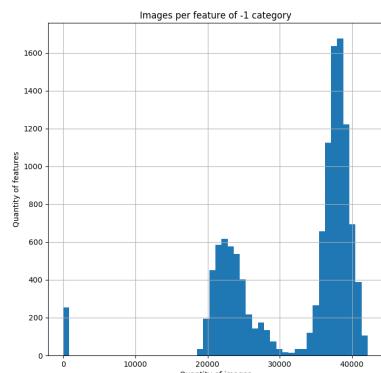
Now I calculate for each feature how many images activate in each specific category:



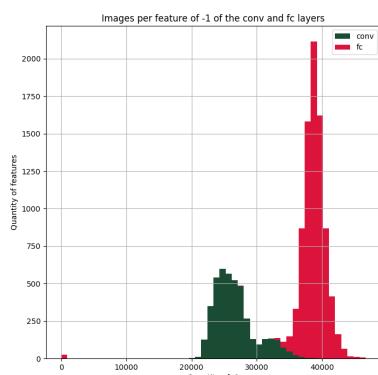
Embedding 19



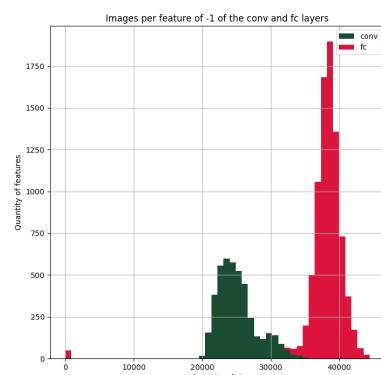
Embedding 25



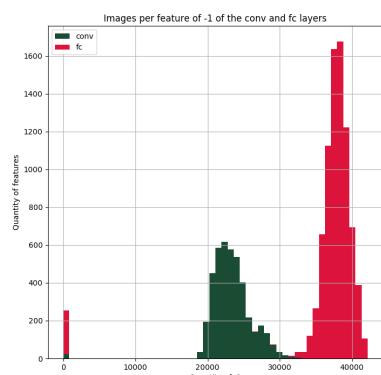
Embedding 31



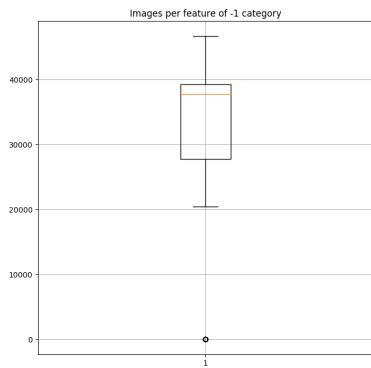
Embedding 19



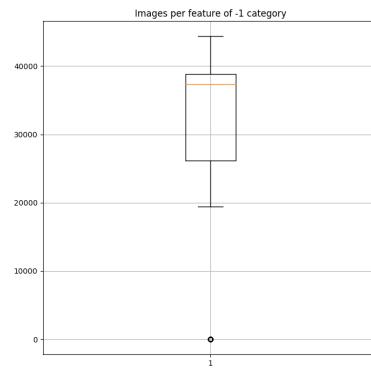
Embedding 25



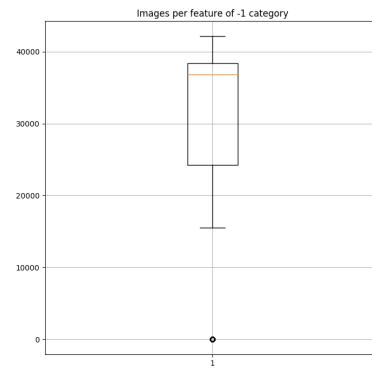
Embedding 31



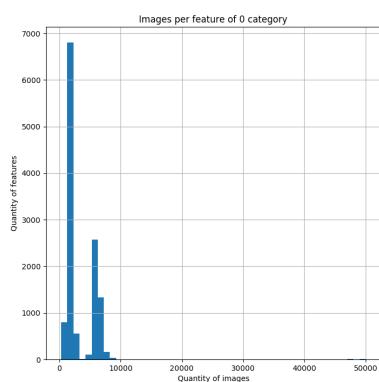
Embedding 19



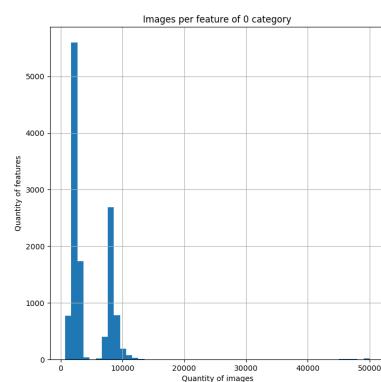
Embedding 25



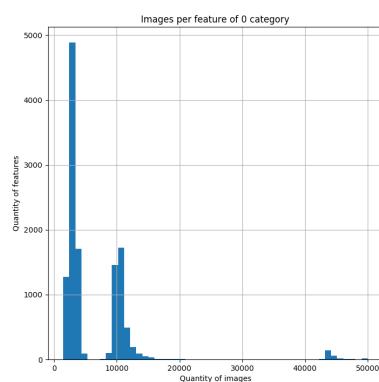
Embedding 31



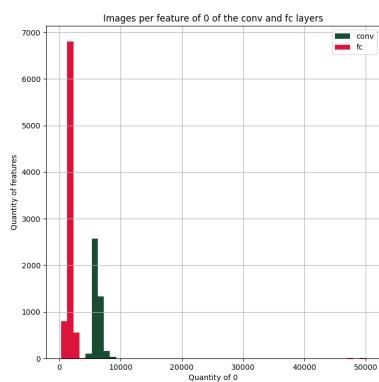
Embedding 19



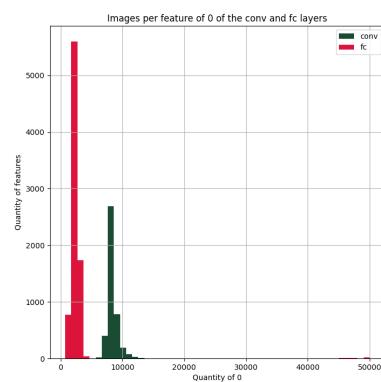
Embedding 25



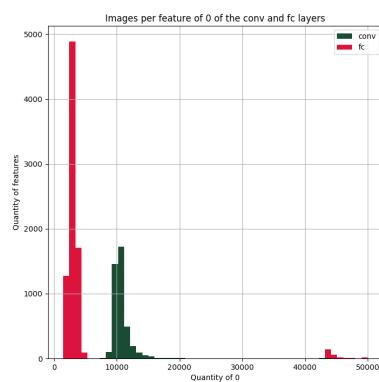
Embedding 31



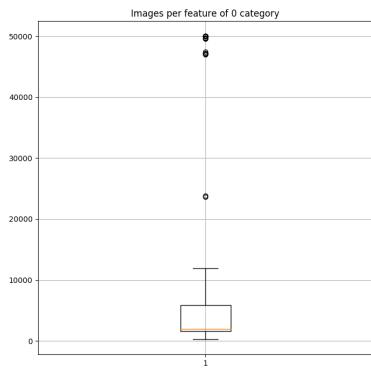
Embedding 19



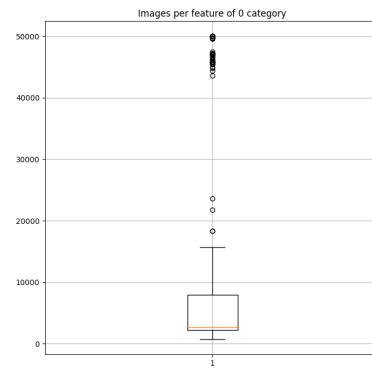
Embedding 25



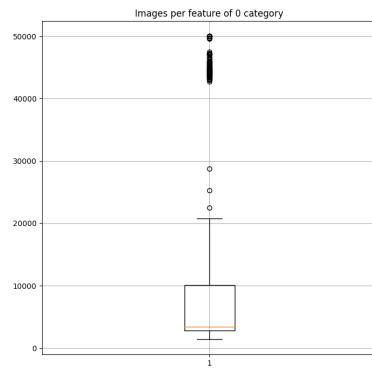
Embedding 31



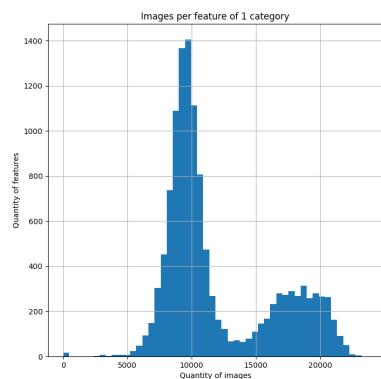
Embedding 19



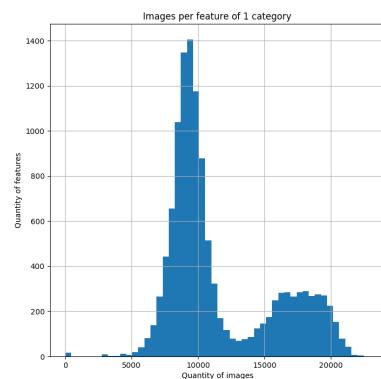
Embedding 25



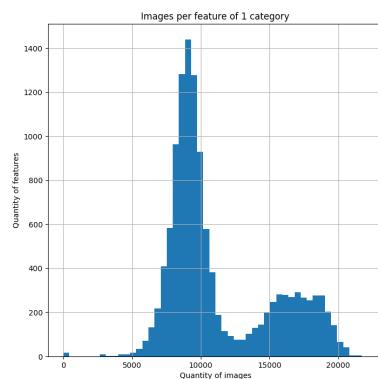
Embedding 31



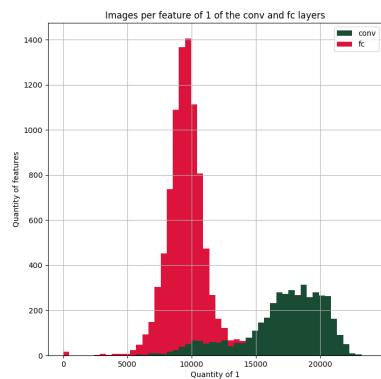
Embedding 19



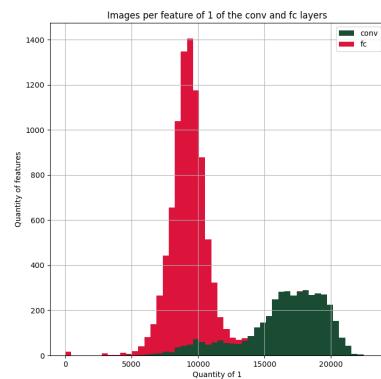
Embedding 25



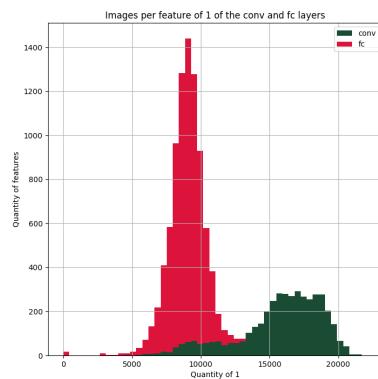
Embedding 31



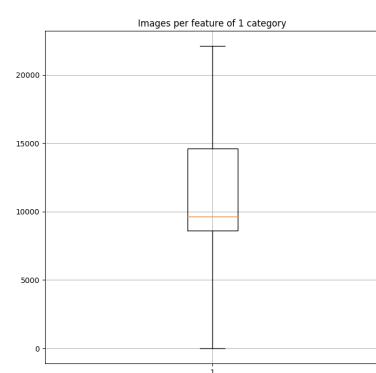
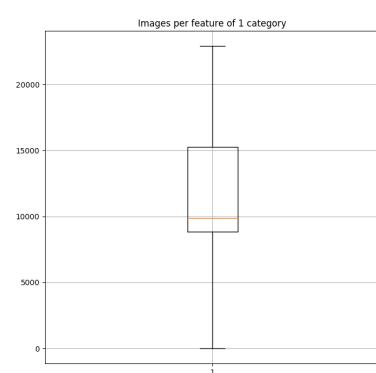
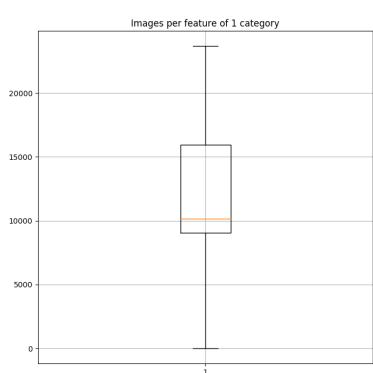
Embedding 19



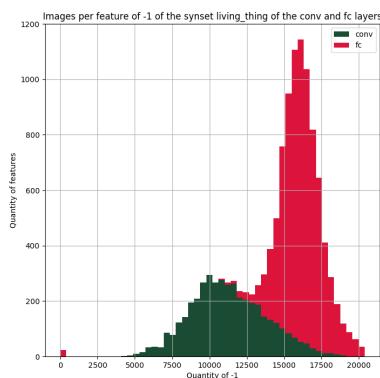
Embedding 25



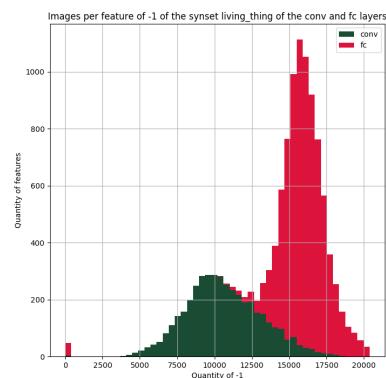
Embedding 31



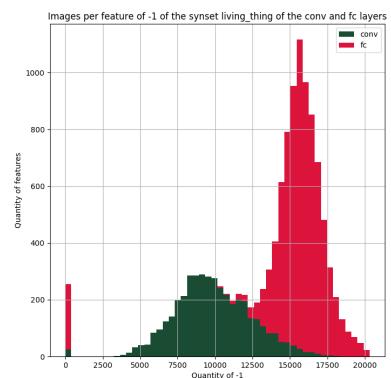
### 3.7 Images per feature per synset



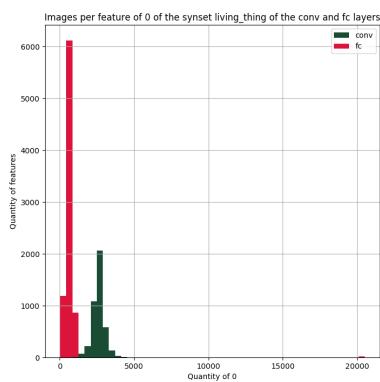
Embedding 19



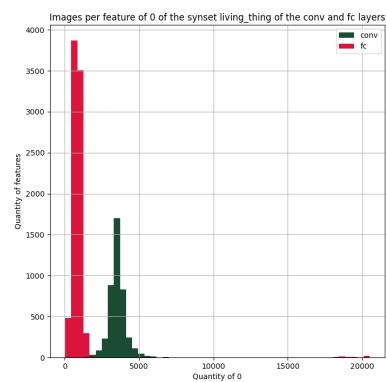
Embedding 25



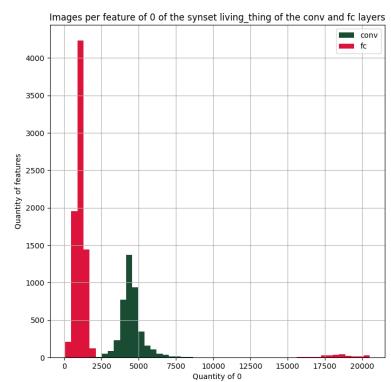
Embedding 31



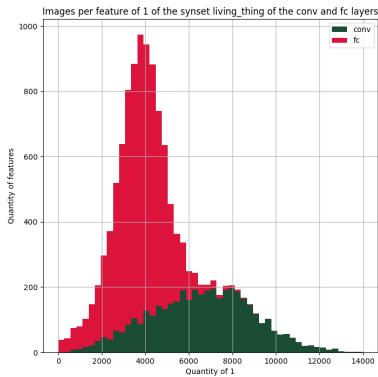
Embedding 19



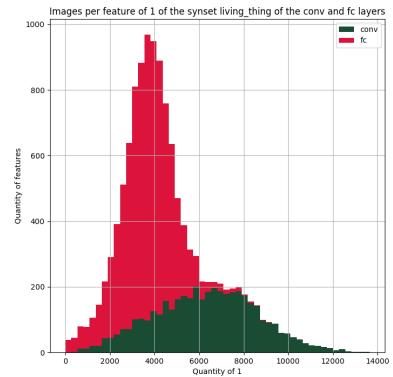
Embedding 25



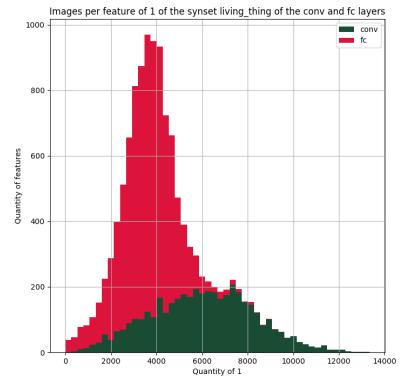
Embedding 31



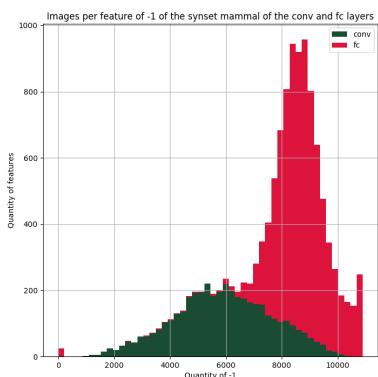
Embedding 19



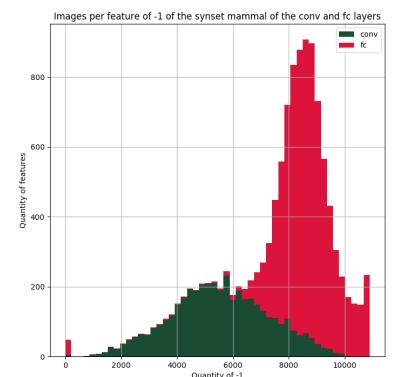
Embedding 25



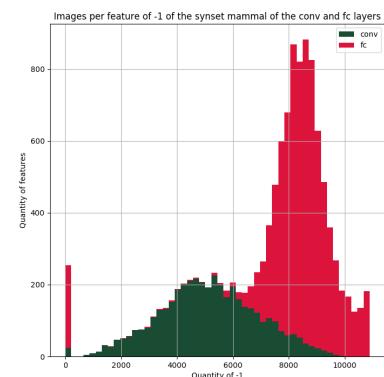
Embedding 31



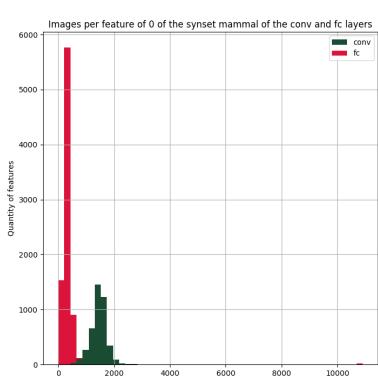
Embedding 19



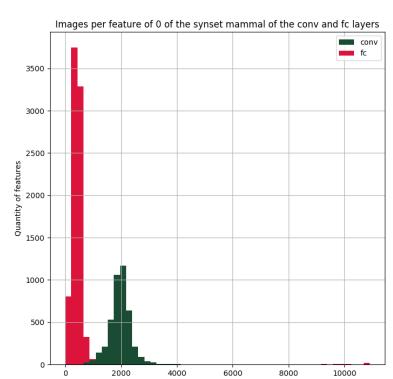
Embedding 25



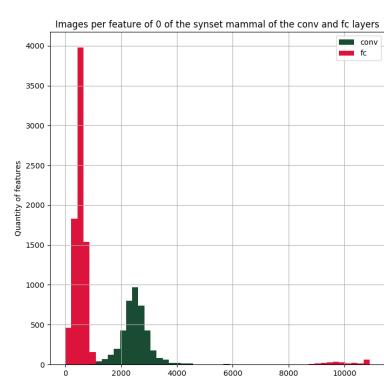
Embedding 31



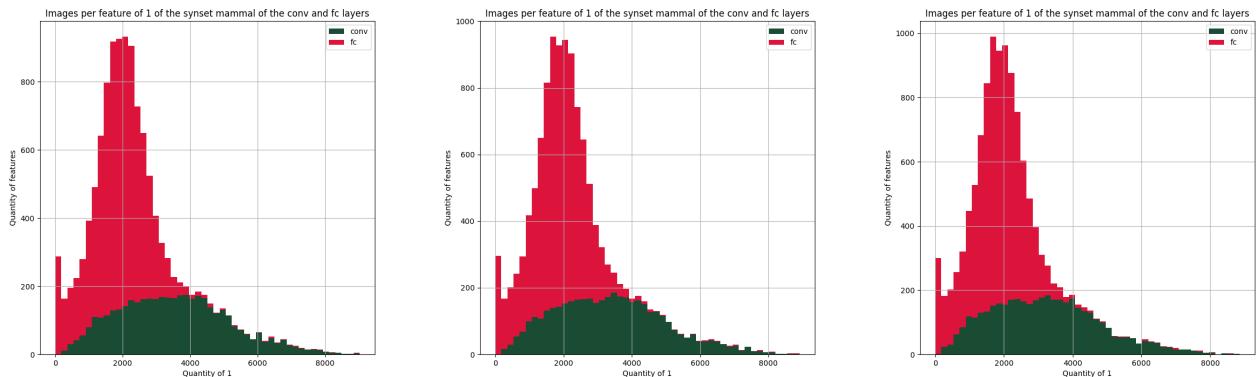
Embedding 19



Embedding 25



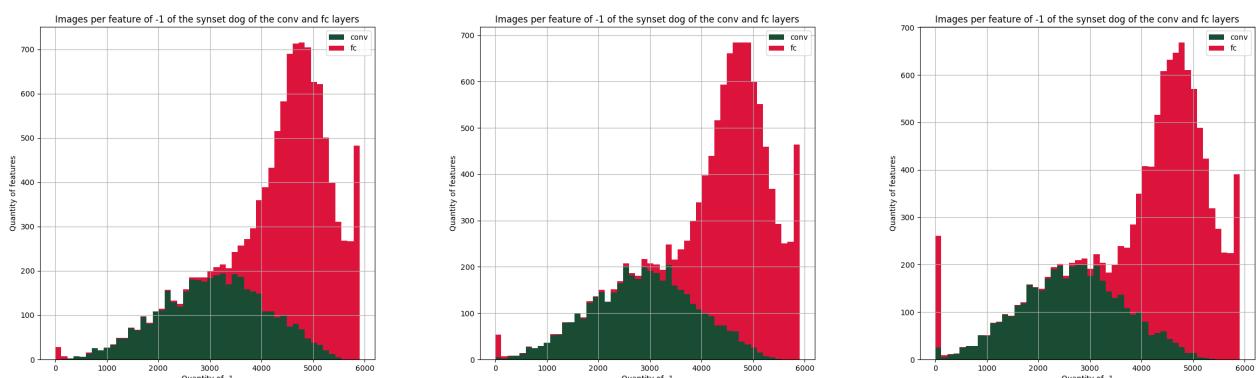
Embedding 31



Embedding 19

Embedding 25

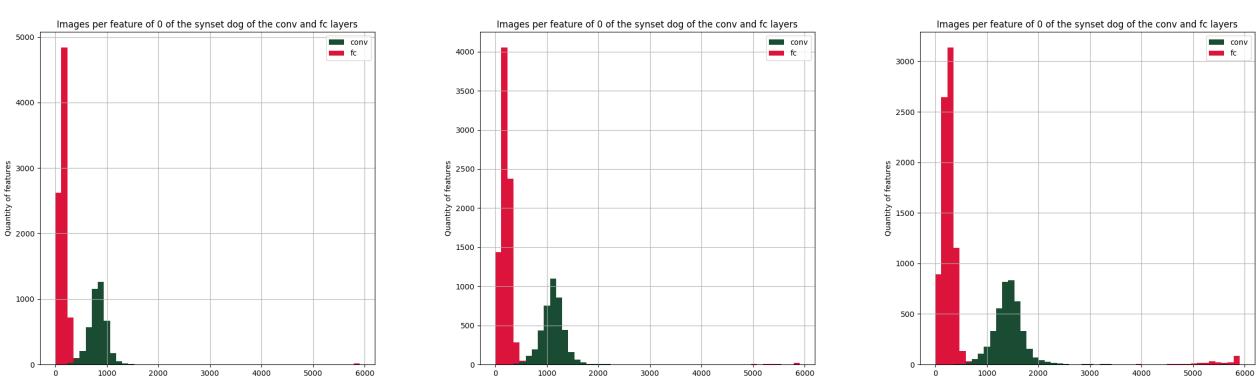
Embedding 31



Embedding 19

Embedding 25

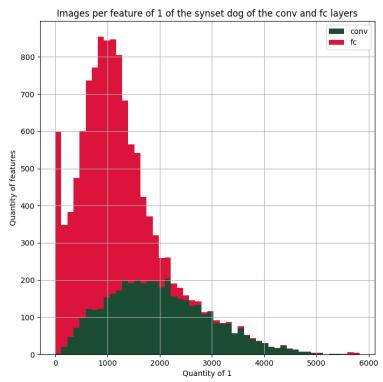
Embedding 31



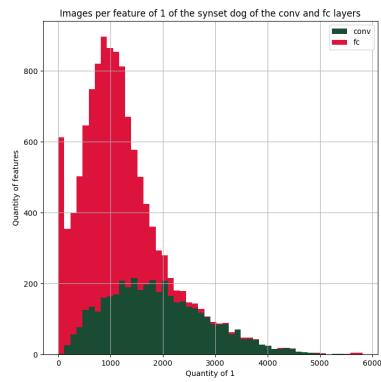
Embedding 19

Embedding 25

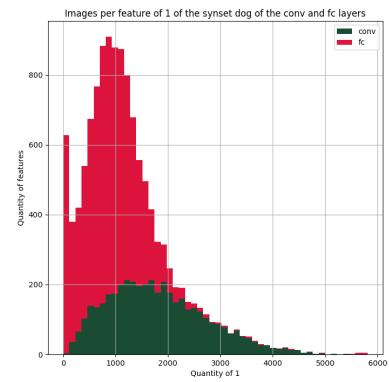
Embedding 31



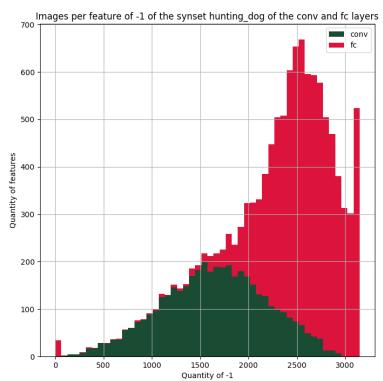
Embedding 19



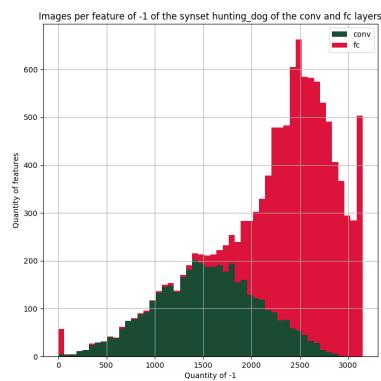
Embedding 25



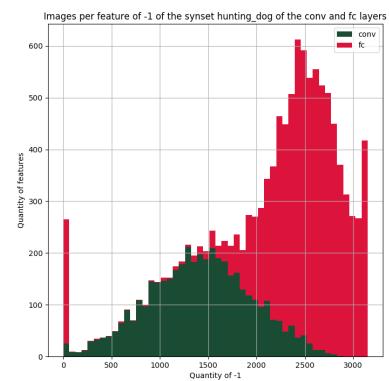
Embedding 31



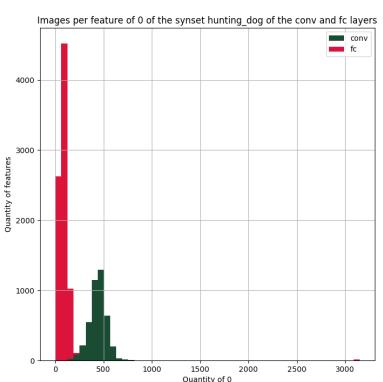
Embedding 19



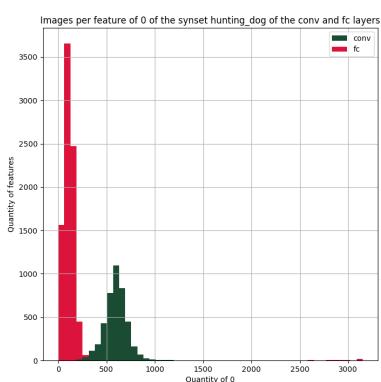
Embedding 25



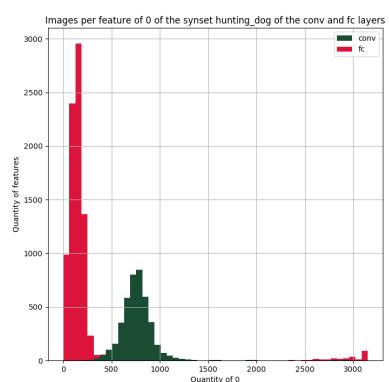
Embedding 31



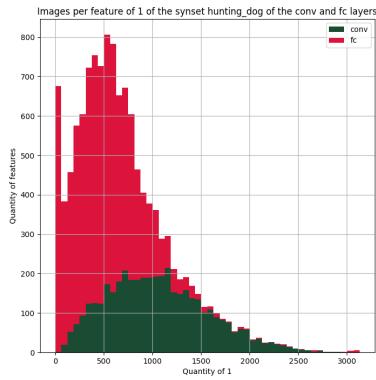
Embedding 19



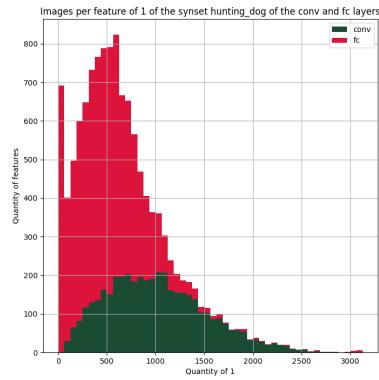
Embedding 25



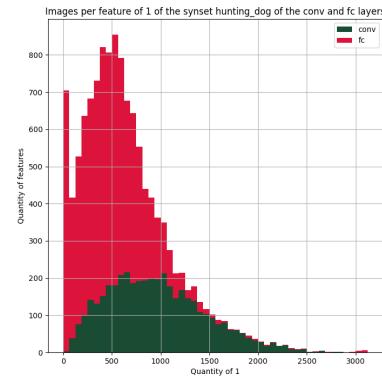
Embedding 31



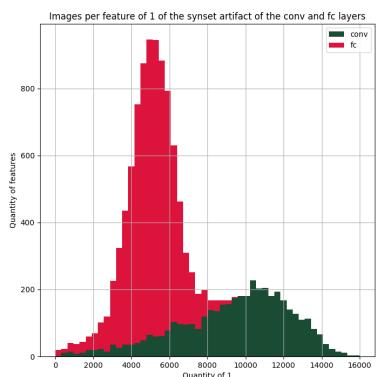
Embedding 19



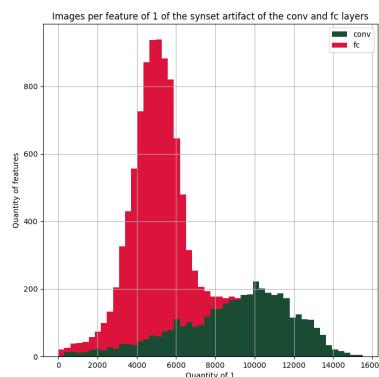
Embedding 25



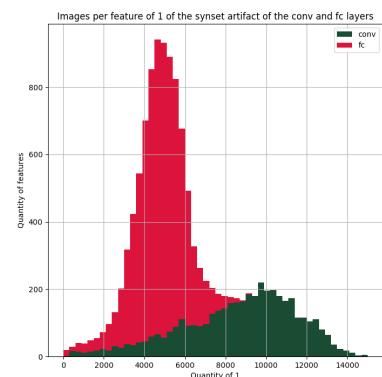
Embedding 31



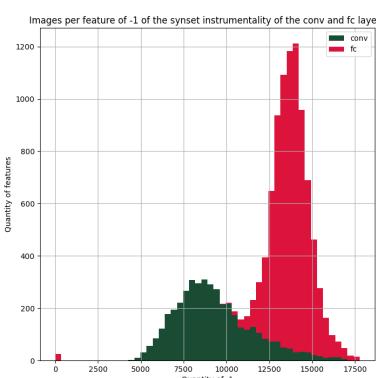
Embedding 19



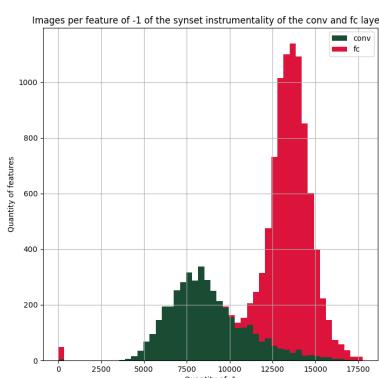
Embedding 25



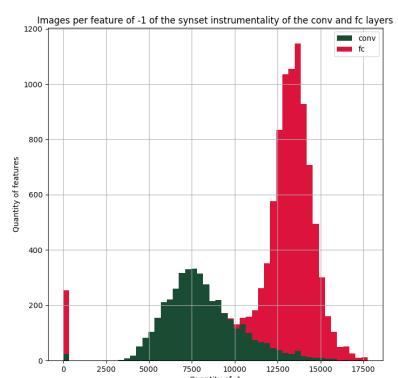
Embedding 31



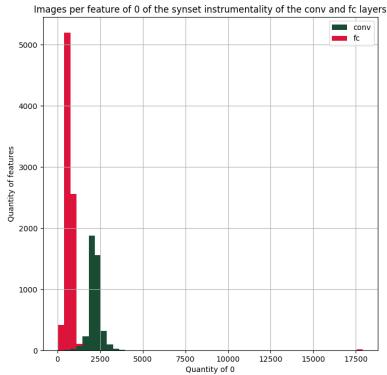
Embedding 19



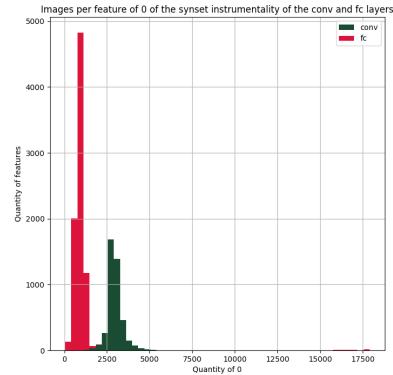
Embedding 25



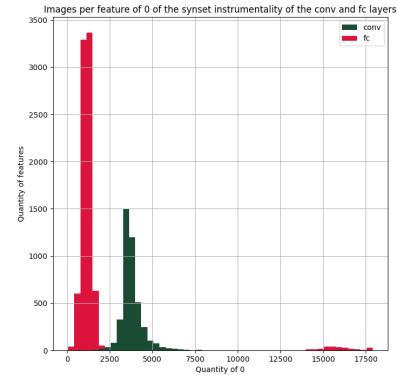
Embedding 31



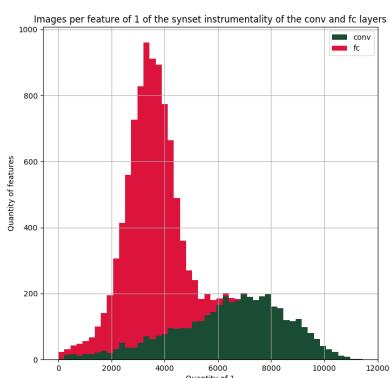
Embedding 19



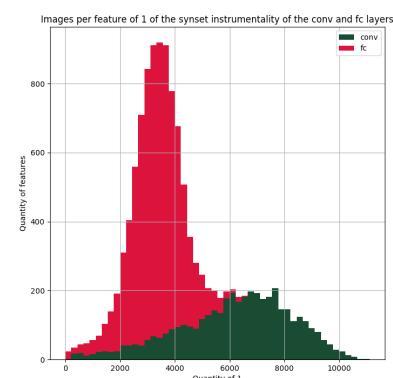
Embedding 25



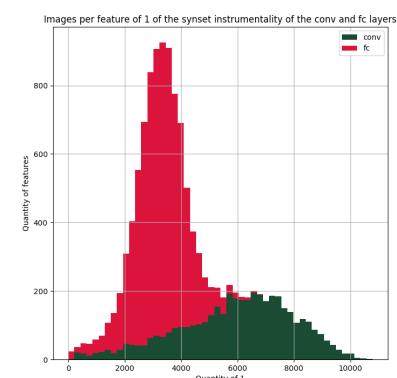
Embedding 31



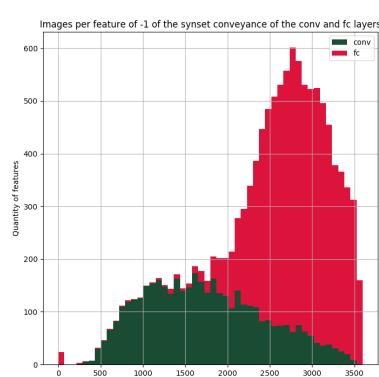
Embedding 19



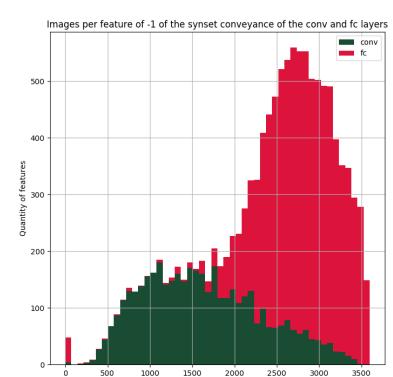
Embedding 25



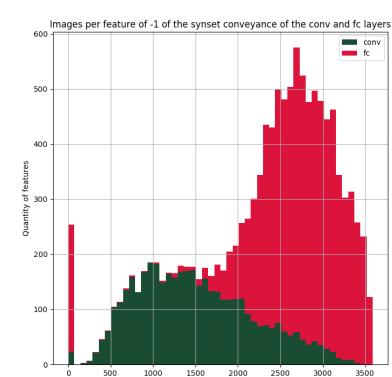
Embedding 31



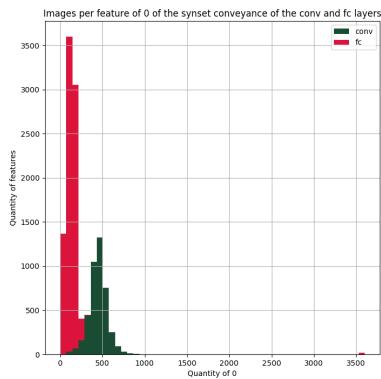
Embedding 19



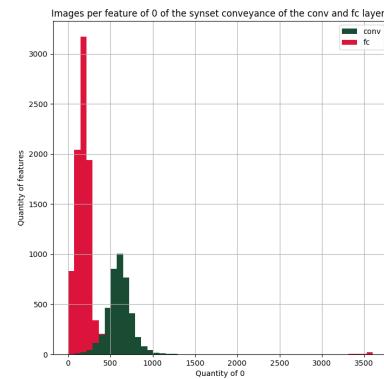
Embedding 25



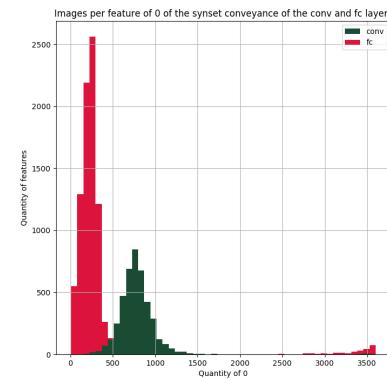
Embedding 31



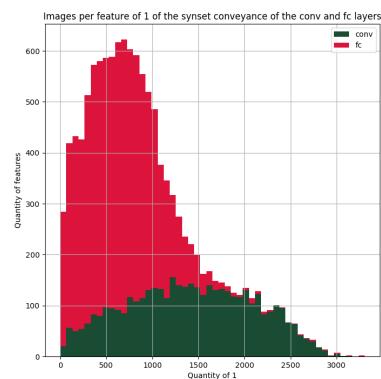
Embedding 19



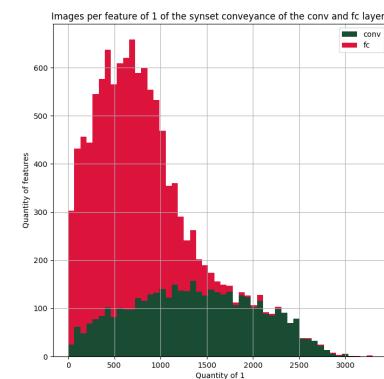
Embedding 25



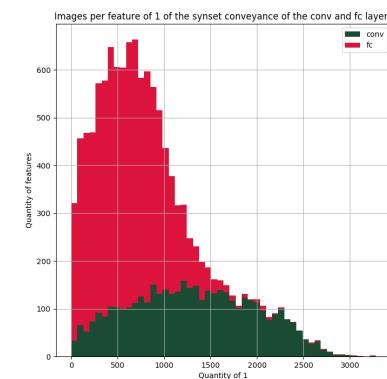
Embedding 31



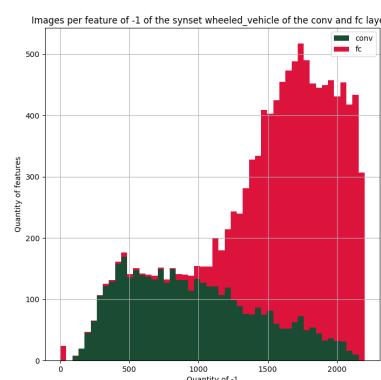
Embedding 19



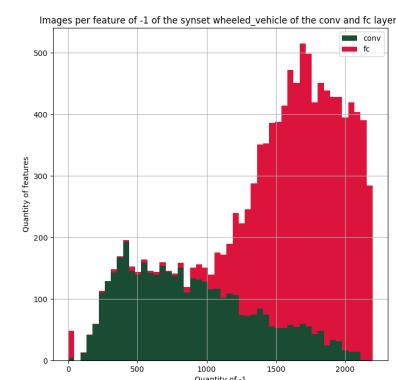
Embedding 25



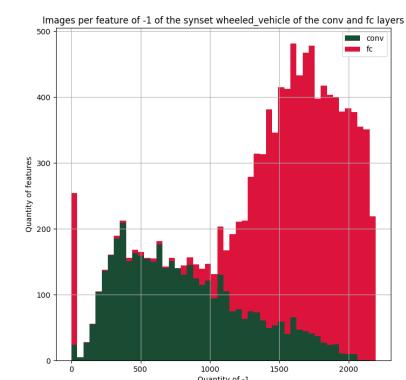
Embedding 31



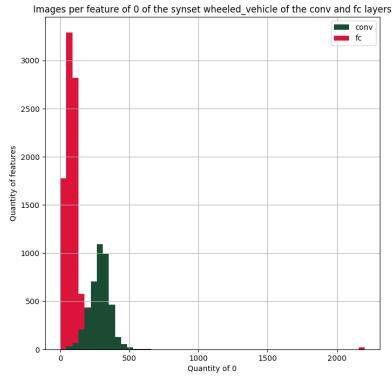
Embedding 19



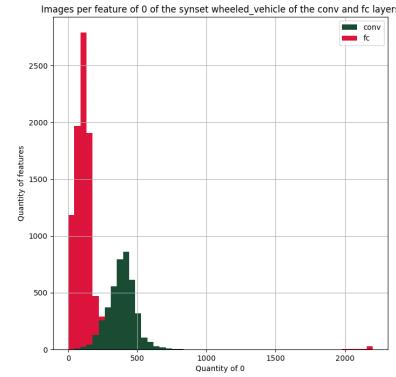
Embedding 25



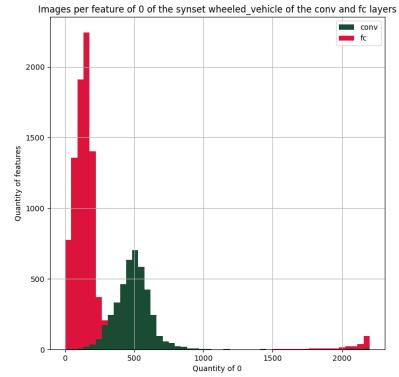
Embedding 31



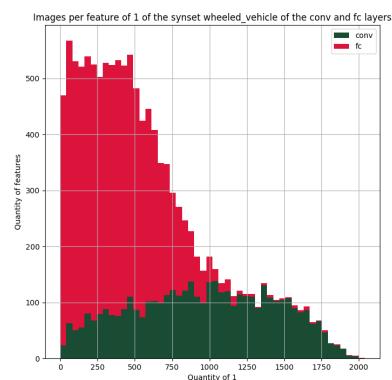
Embedding 19



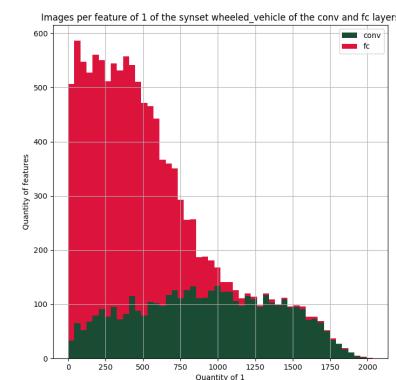
Embedding 25



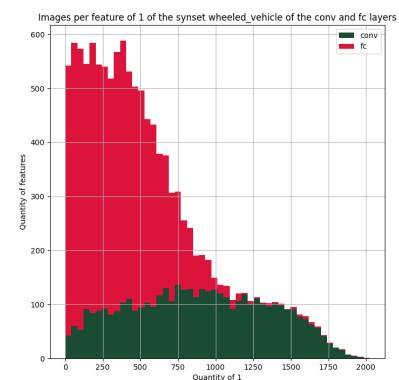
Embedding 31



Embedding 19



Embedding 25



Embedding 31

### 3.8 Comprobación de que las cosas tienen sentido

- Hay alguna imagen que no tenga ninguna feature con valor cero? No, ninguna
- Hay algún synset que tenga valor 1 y -1 para la misma feature? Si, bastantes. Estamos tomando:  
La feature tiene el valor i para el synset si lo tiene para alguna imagen del synset(Estamos estudiando la submatriz correspondiente al synset)

### 3.9 Estudio de los outliers de imágenes por feature

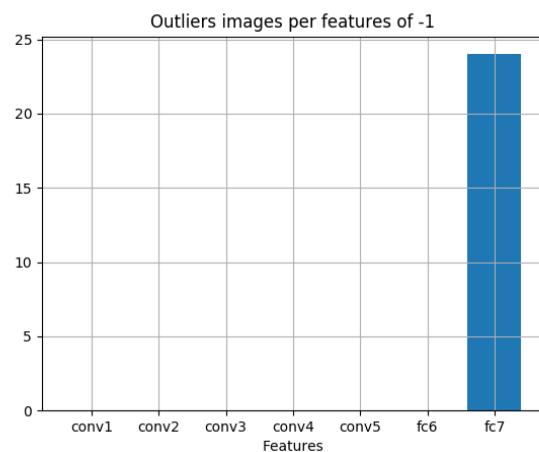


Figure 48: Category: -1

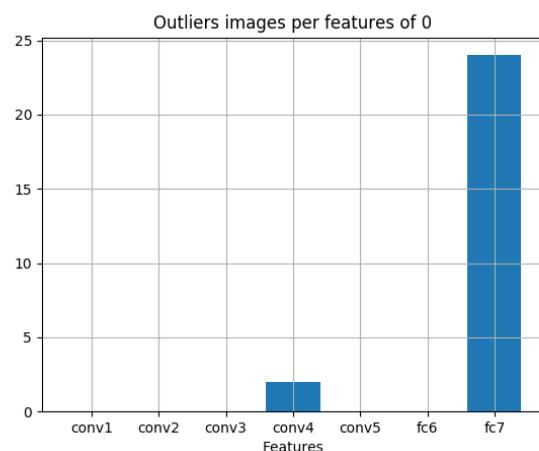


Figure 49: Category: 0

### 3.10 Matrices de cambio

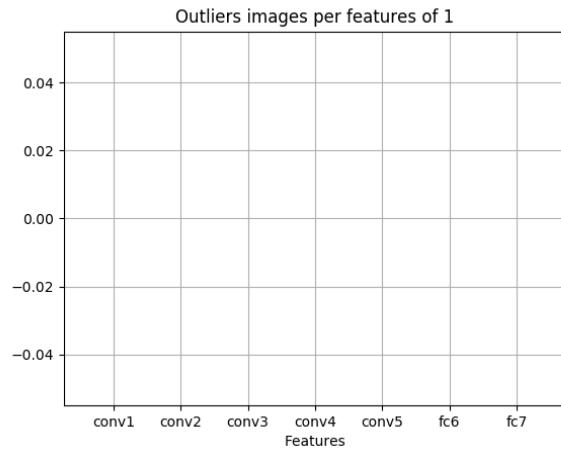
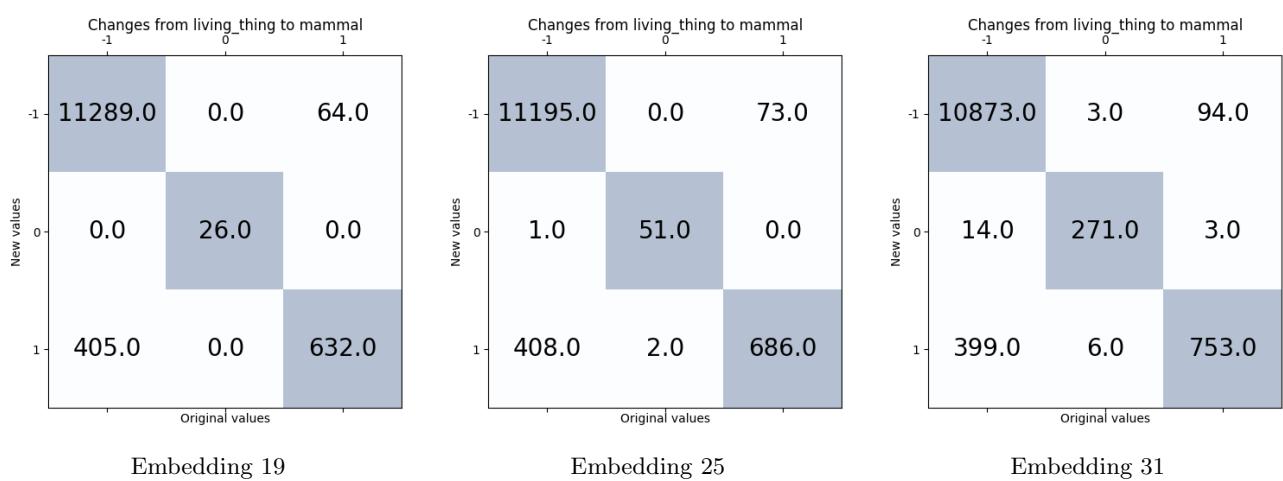
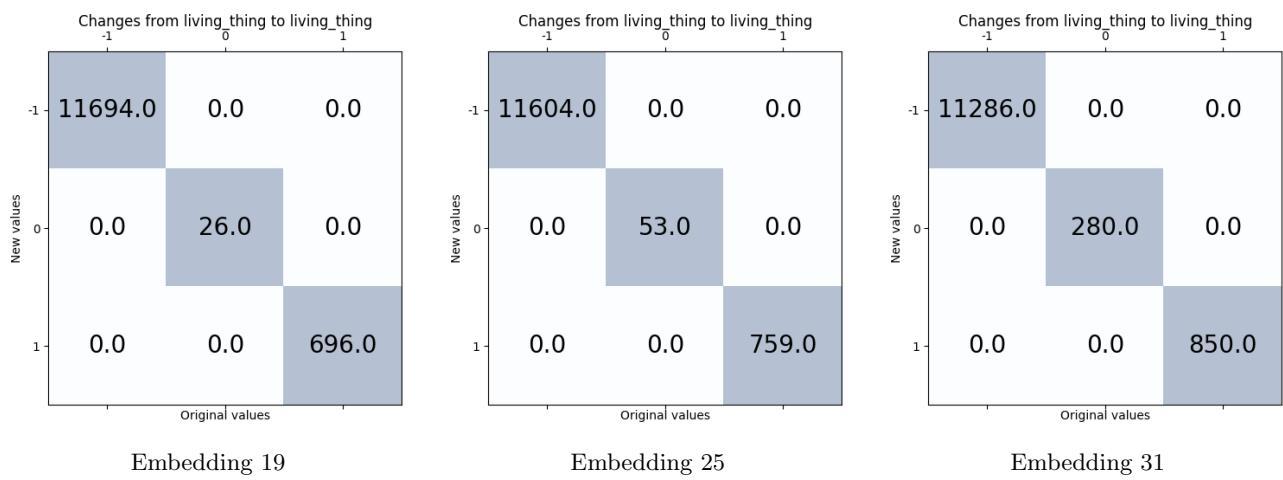
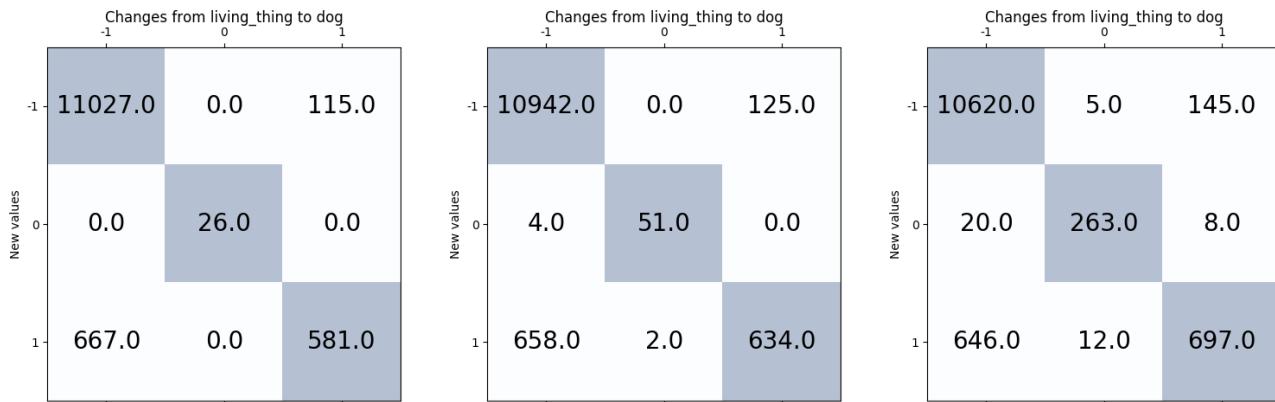


Figure 50: Category: 1





Embedding 19

Embedding 25

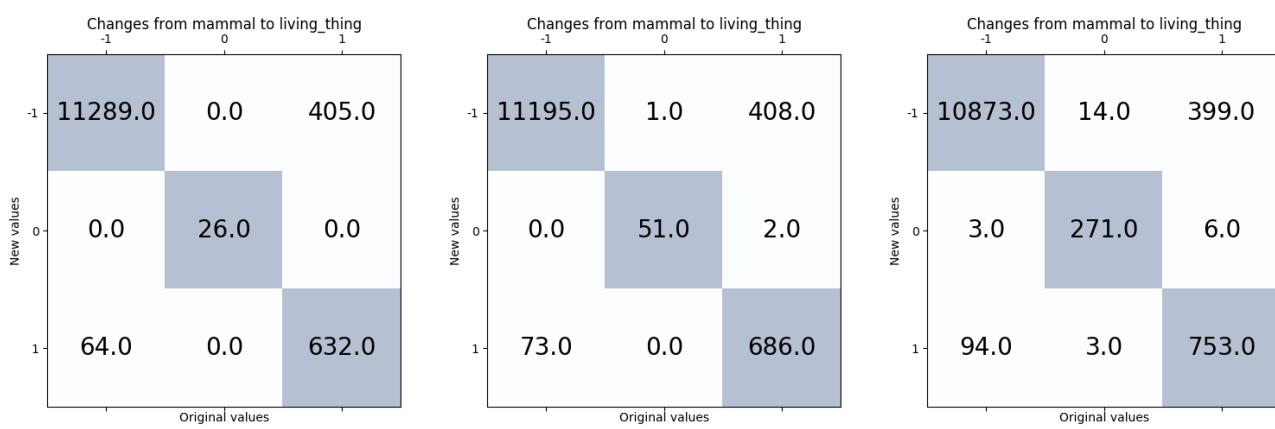
Embedding 31



Embedding 19

Embedding 25

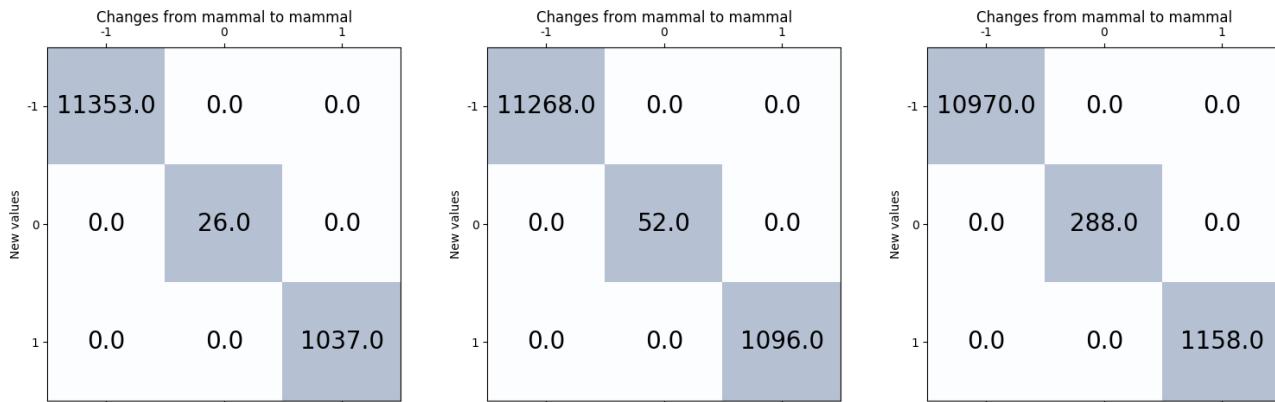
Embedding 31



Embedding 19

Embedding 25

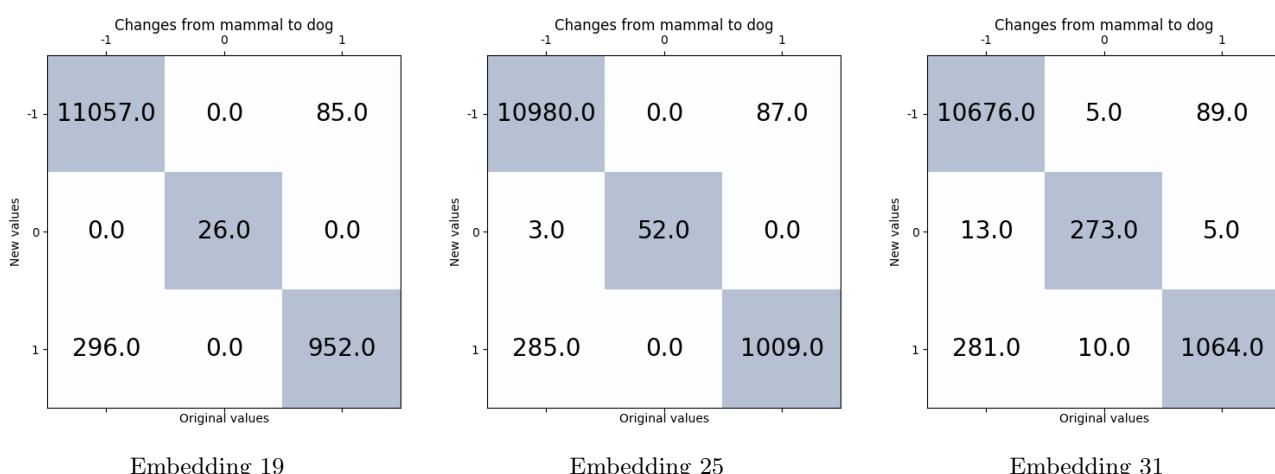
Embedding 31



Embedding 19

Embedding 25

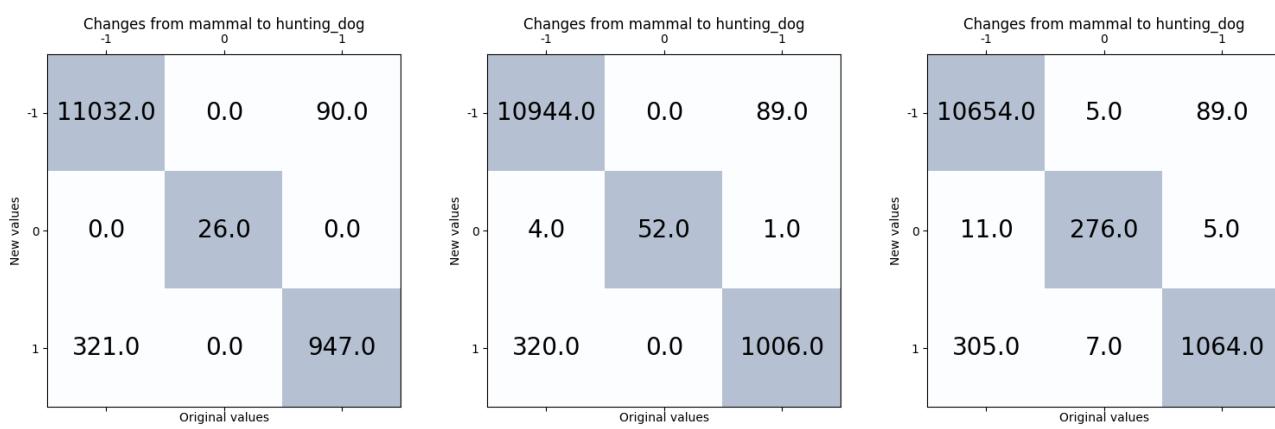
Embedding 31



Embedding 19

Embedding 25

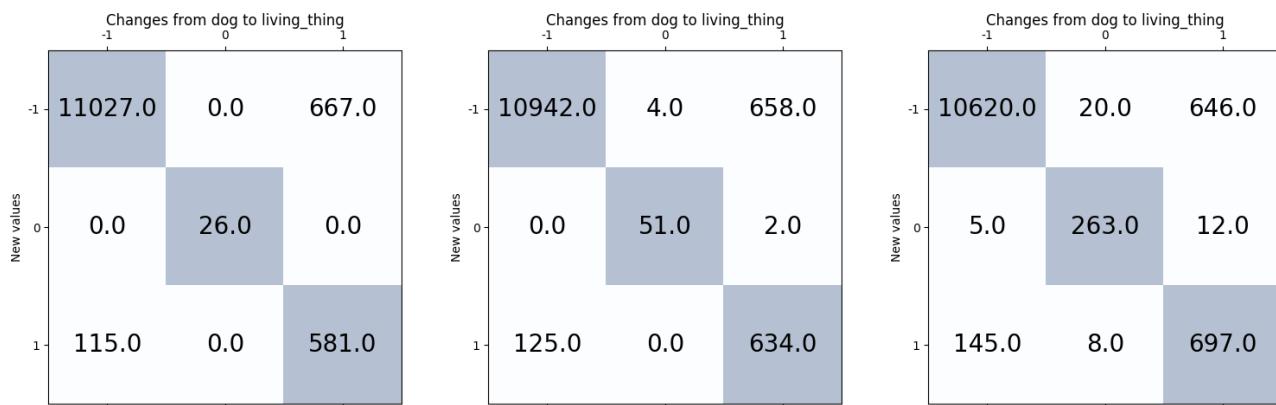
Embedding 31



Embedding 19

Embedding 25

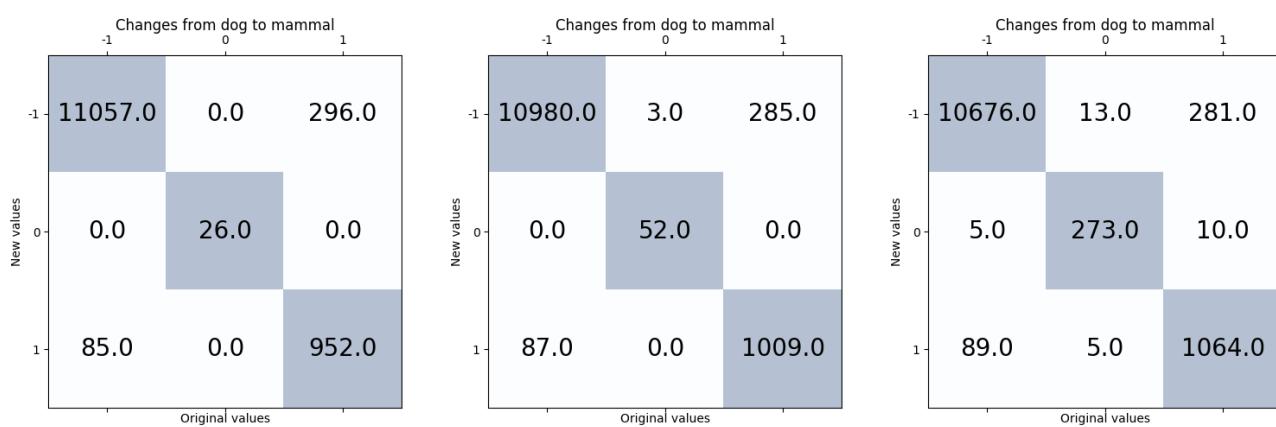
Embedding 31



Embedding 19

Embedding 25

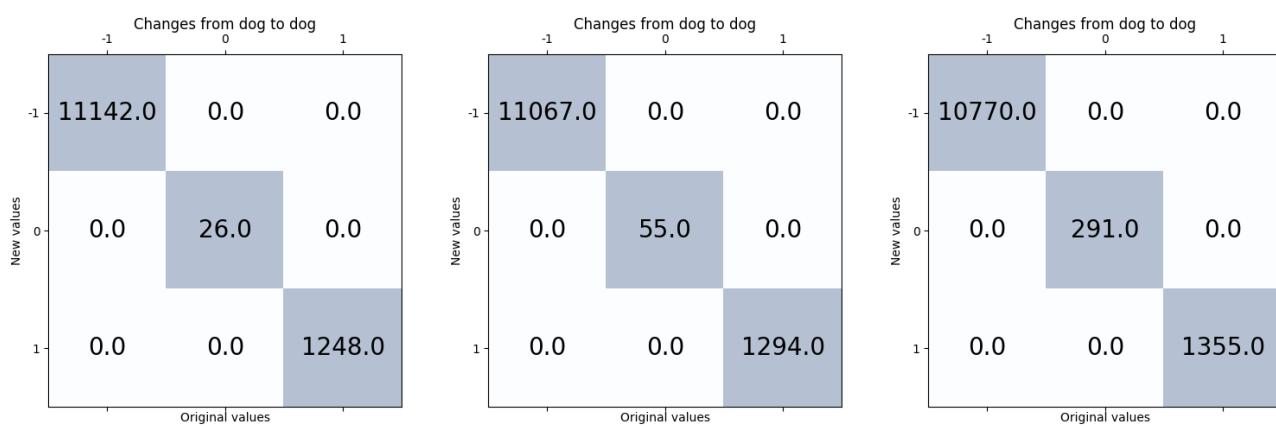
Embedding 31



Embedding 19

Embedding 25

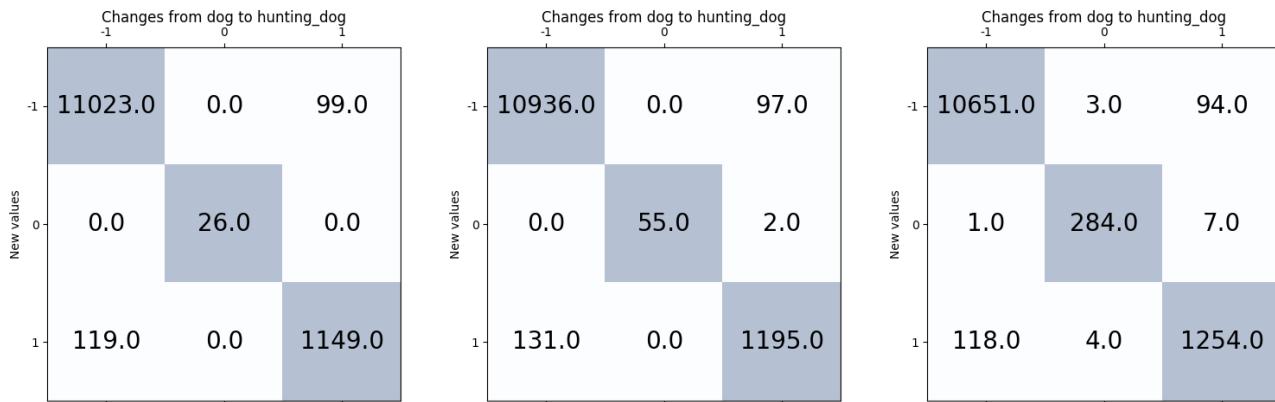
Embedding 31



Embedding 19

Embedding 25

Embedding 31



Embedding 19

Embedding 25

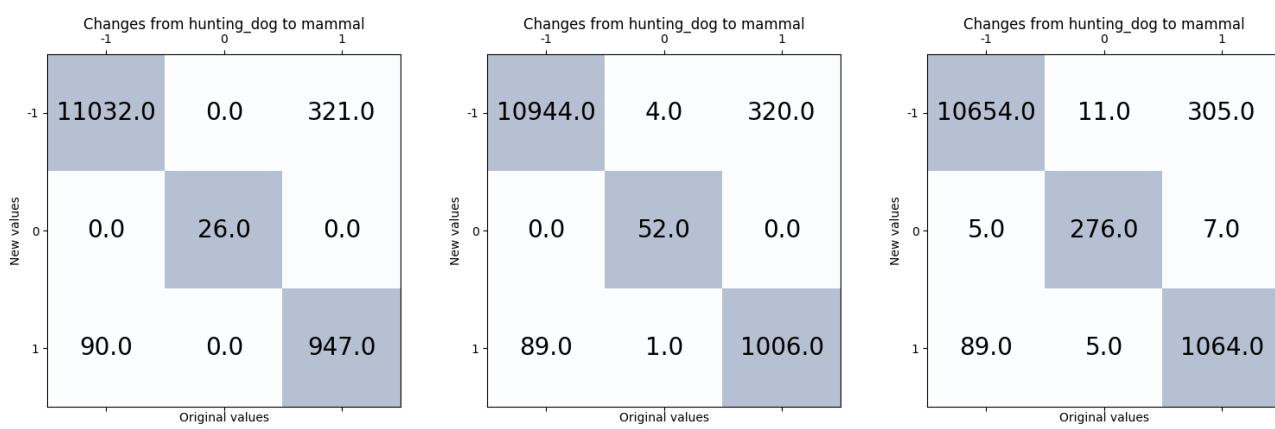
Embedding 31



Embedding 19

Embedding 25

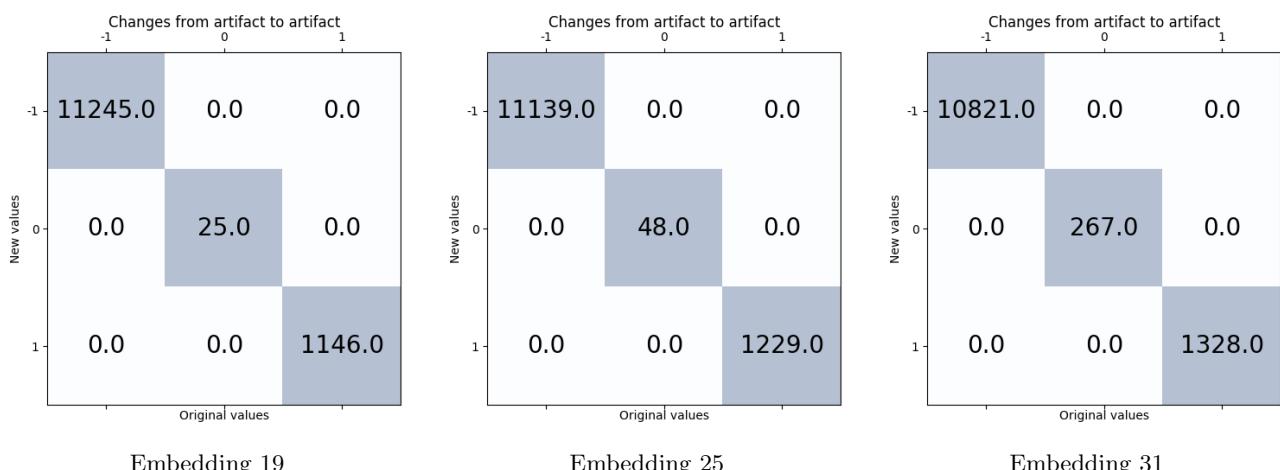
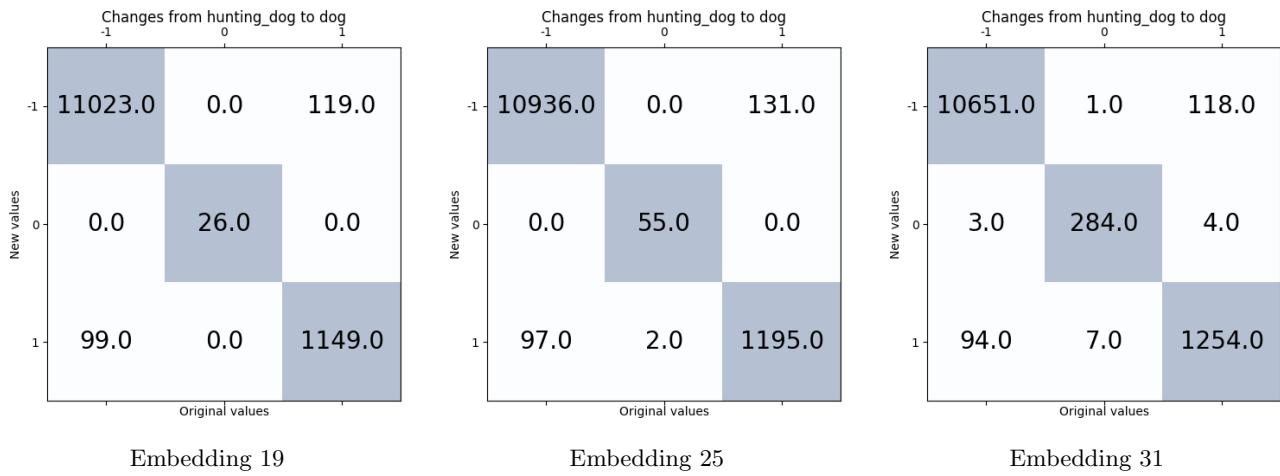
Embedding 31

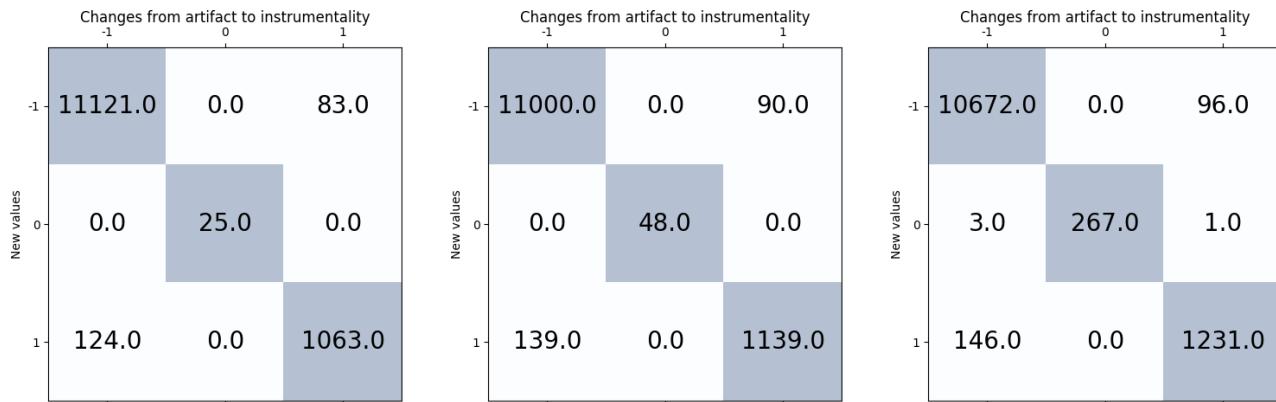


Embedding 19

Embedding 25

Embedding 31

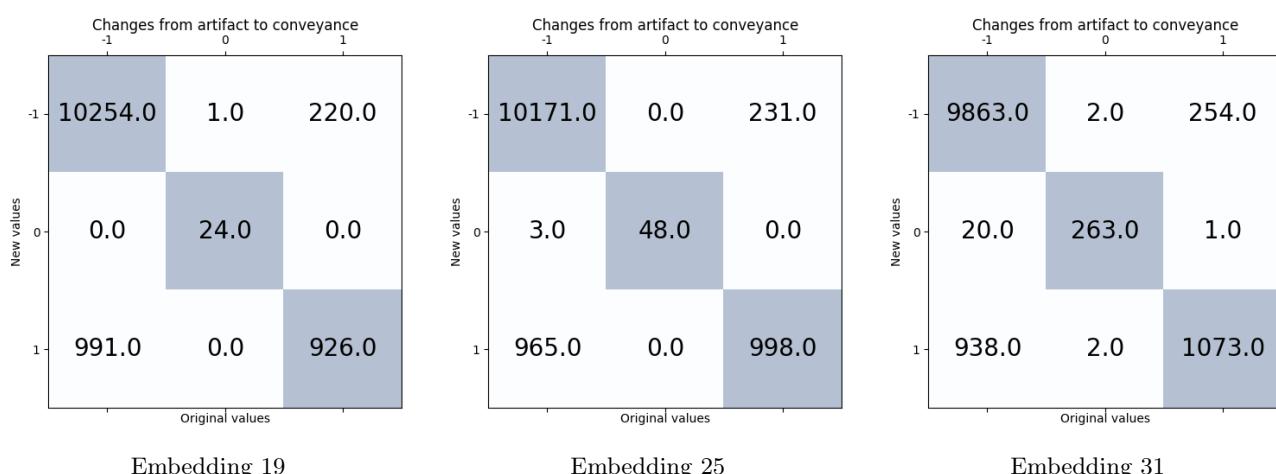




Embedding 19

Embedding 25

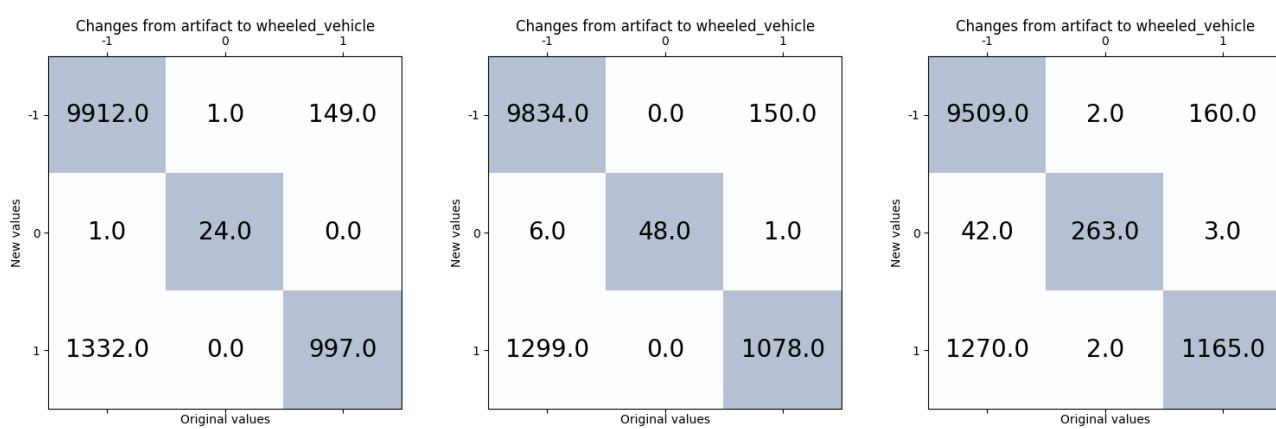
Embedding 31



Embedding 19

Embedding 25

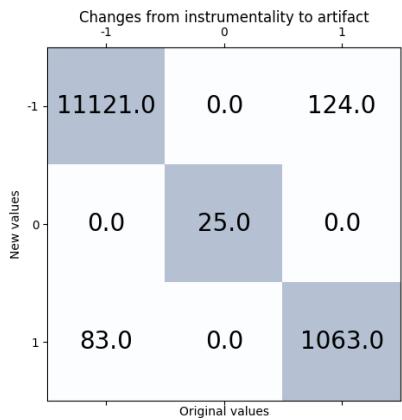
Embedding 31



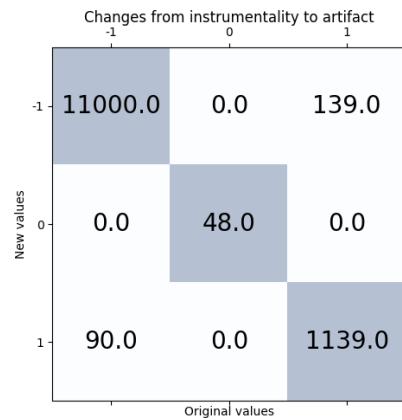
Embedding 19

Embedding 25

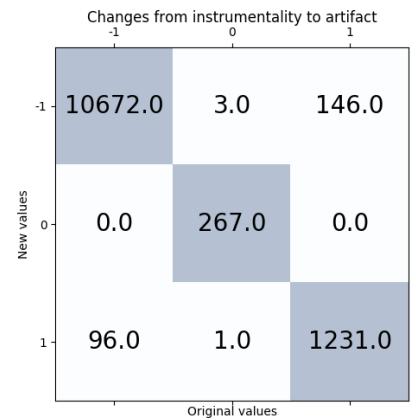
Embedding 31



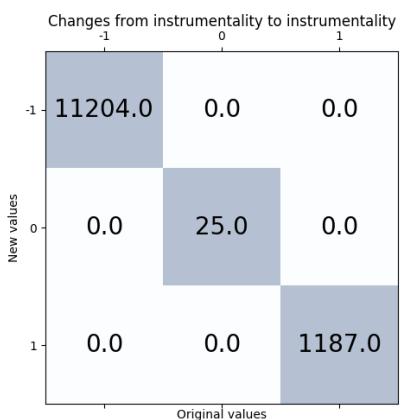
Embedding 19



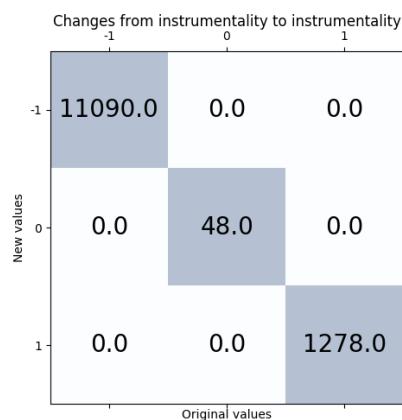
Embedding 25



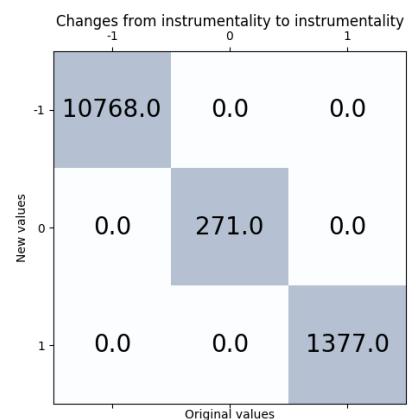
Embedding 31



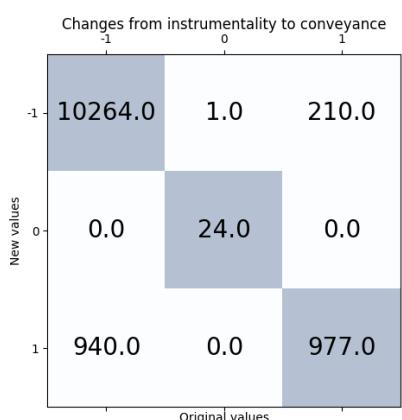
Embedding 19



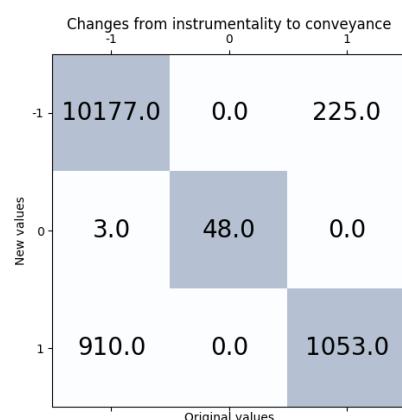
Embedding 25



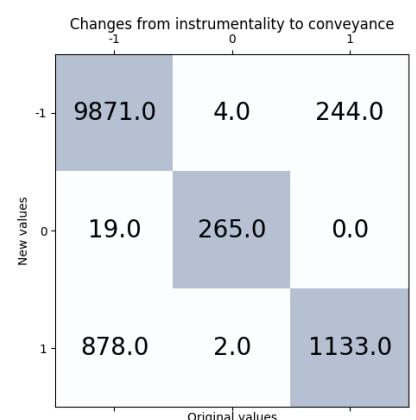
Embedding 31



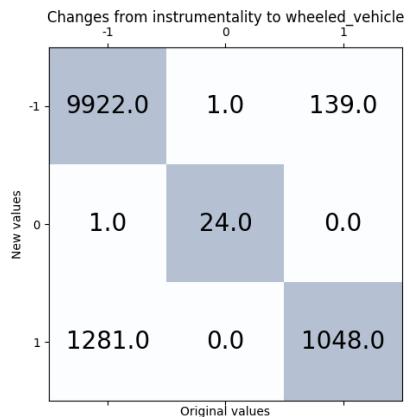
Embedding 19



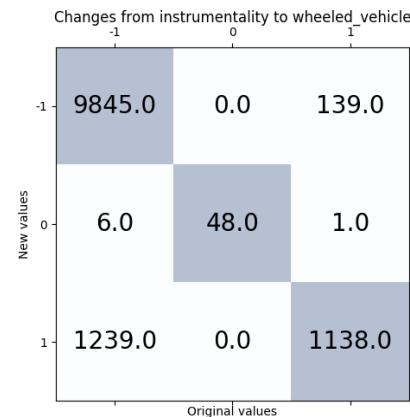
Embedding 25



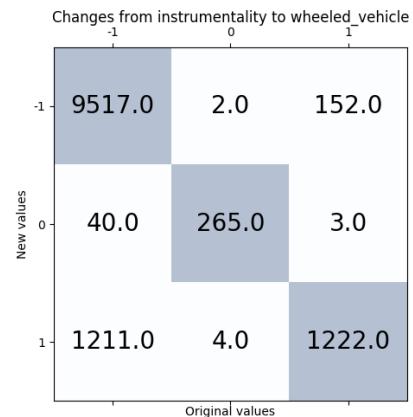
Embedding 31



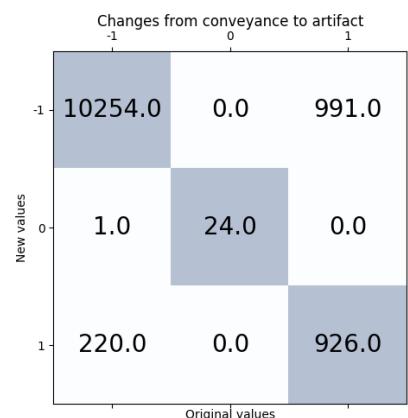
Embedding 19



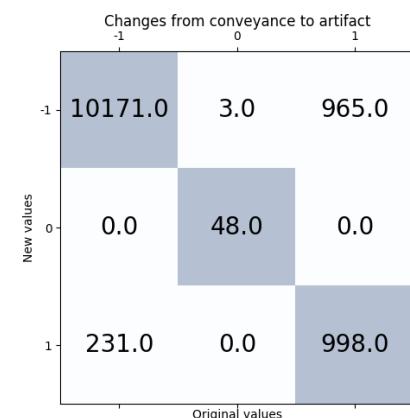
Embedding 25



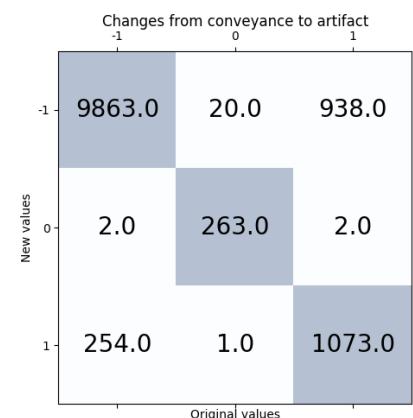
Embedding 31



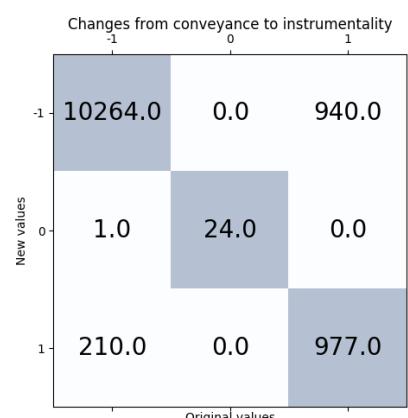
Embedding 19



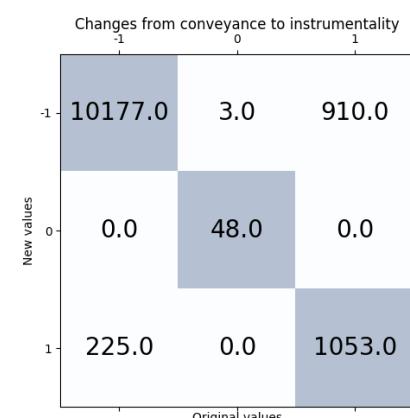
Embedding 25



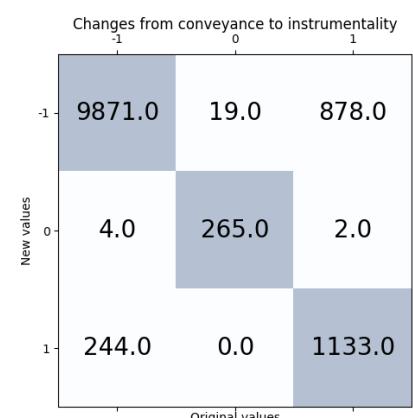
Embedding 31



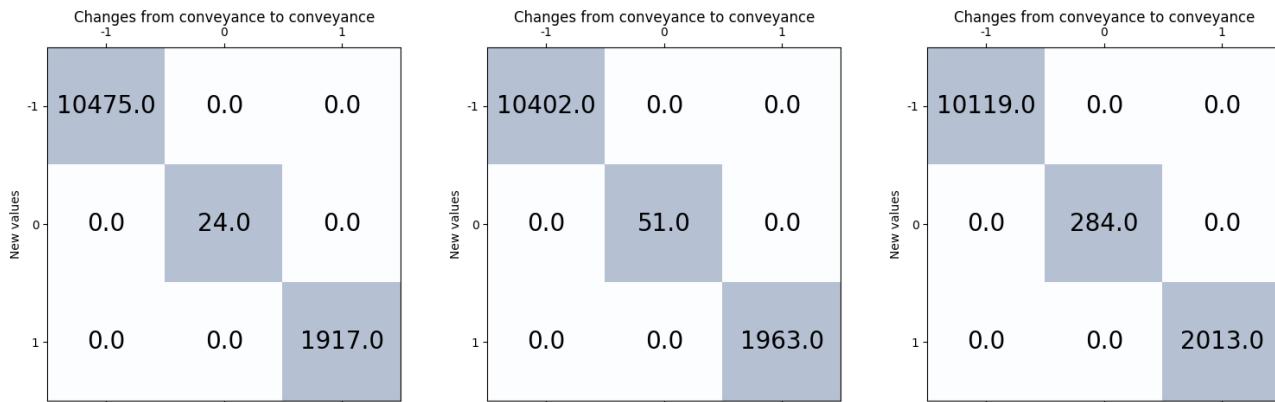
Embedding 19



Embedding 25



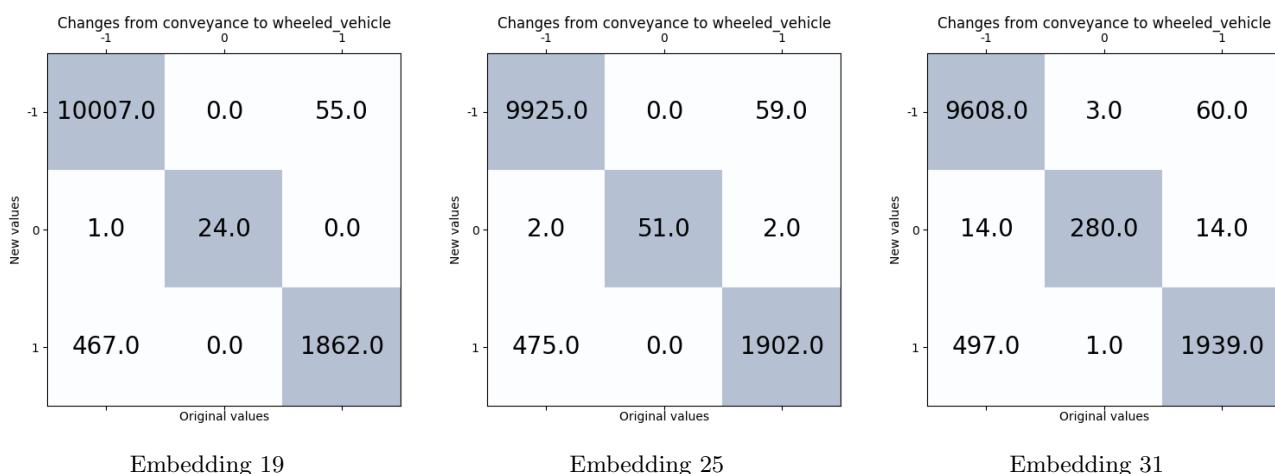
Embedding 31



Embedding 19

Embedding 25

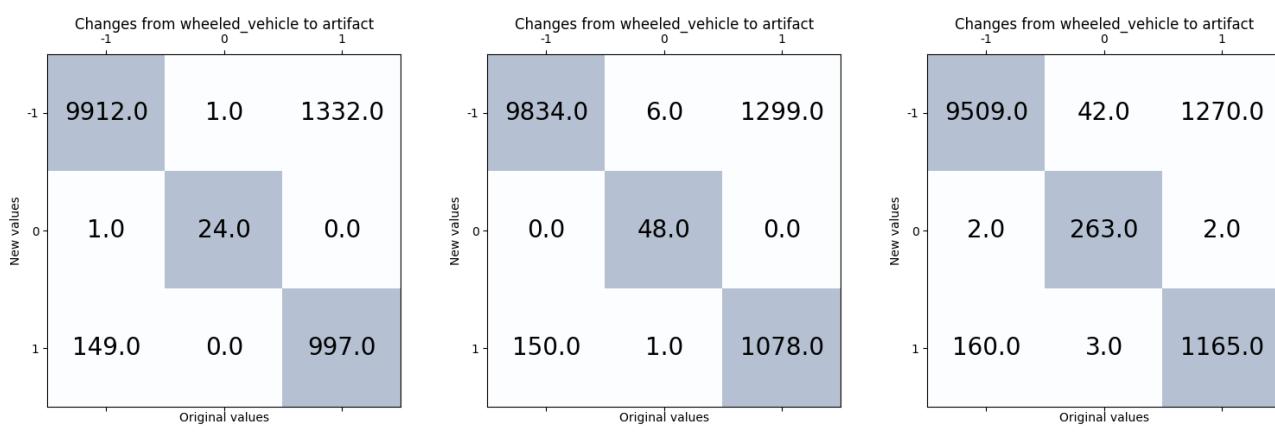
Embedding 31



Embedding 19

Embedding 25

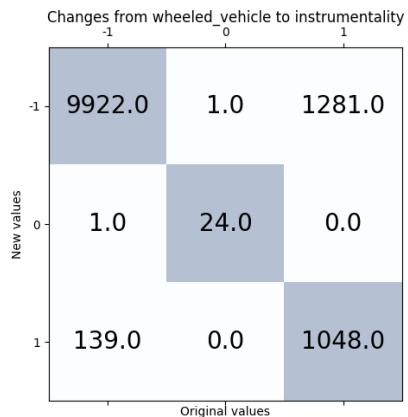
Embedding 31



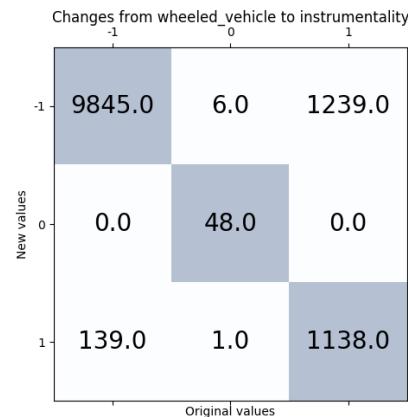
Embedding 19

Embedding 25

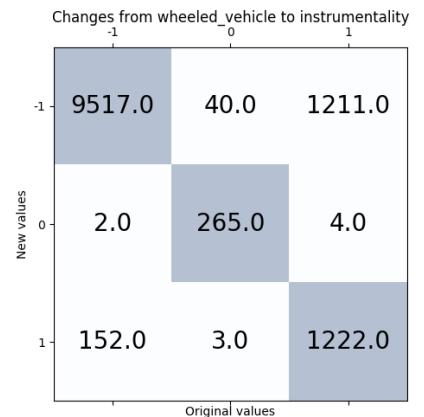
Embedding 31



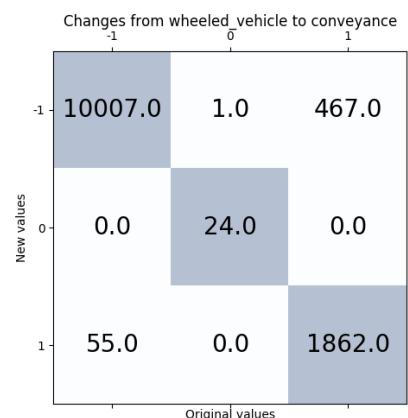
Embedding 19



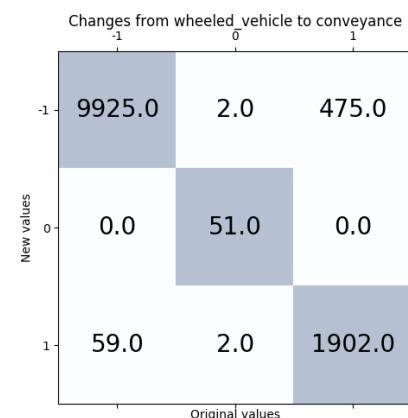
Embedding 25



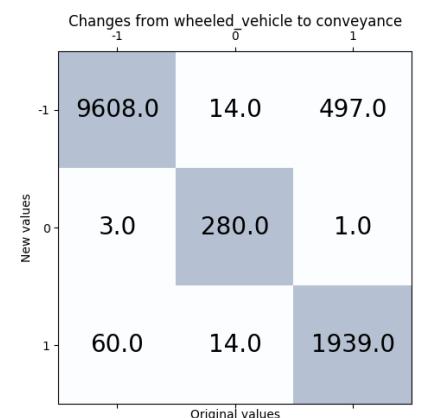
Embedding 31



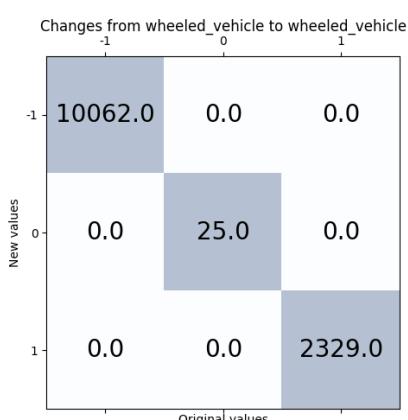
Embedding 19



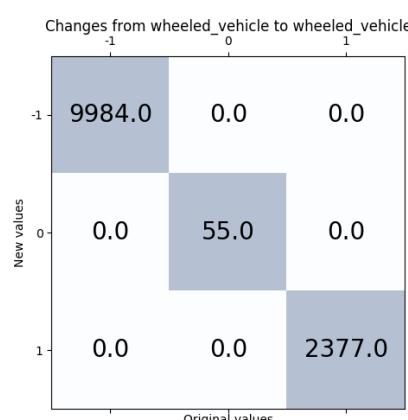
Embedding 25



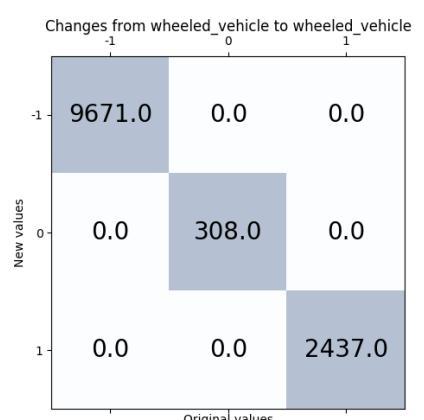
Embedding 31



Embedding 19



Embedding 25



Embedding 31