
	Politechnika Bydgoska im. J. J. Śniadeckich Wydział Telekomunikacji, Informatyki i Elektrotechniki Zakład Systemów Teleinformatycznych		
Przedmiot	Zaawansowane Techniki Sztucznej Inteligencji		
Prowadzący	prof. dr hab. inż. prof. PBS Piotr Cofta		
Temat	<i>Project</i>		
Student	Cezary Tytko		
Ocena		Data oddania spr.	

Etap 1. – kontrola danych.

Weryfikacji możemy dokonać przez wczytanie danych jako pliku csv i sprawdzić czy wymiary otrzymanej tablicy zgadzają się z przewidywanymi, to znaczy czy mamy odpowiednią liczbę rekordów i czy każdy rekord zawiera odpowiednią liczbę danych, to jest suma liczby oczek na widocznych poprawnych kostkach i 100 X 100 pikseli ułożonych w jednym wymiarze. Należy również sprawdzić czy wartości etykiet jak o danych są zgodne z założeniami, np. czy nie wychodzą poza przewidziany zakres, albo czy nie ma wartości brakujących, można to sprawdzić wyświetlając wartości unikatowe (Select distinct w konwencji sql), zgodnie z założeniami wartość etykiety powinna być z zakresu <6, 30>, a dla piksela <0, 255>, jeżeli dane będą zawierały błędy na poziomie typów wartości tzn. string nie konwertowany na int, to dostaniemy błąd na etapie odczytu pliku csv (przynajmniej dla implementacji z pandas).

Przykładowy kod weryfikacji:

```

1. dice_y, dice_x = ReadDiceCSV(dice5_dir)
2. dice_x = dice_x.reshape((dice5_count, 100, -1))
3. print("dice:")
4. print(f"Label Shape: {dice_y.shape} has NAN: {np.isnan(dice_y).any()}")
5. print(f"Values : {np.unique(dice_y)}")
6. print(f"Data Shape: {dice_x.shape} has NAN: {np.isnan(dice_x).any()}")
7. print(f"Values : {np.unique(dice_x)}")
8.

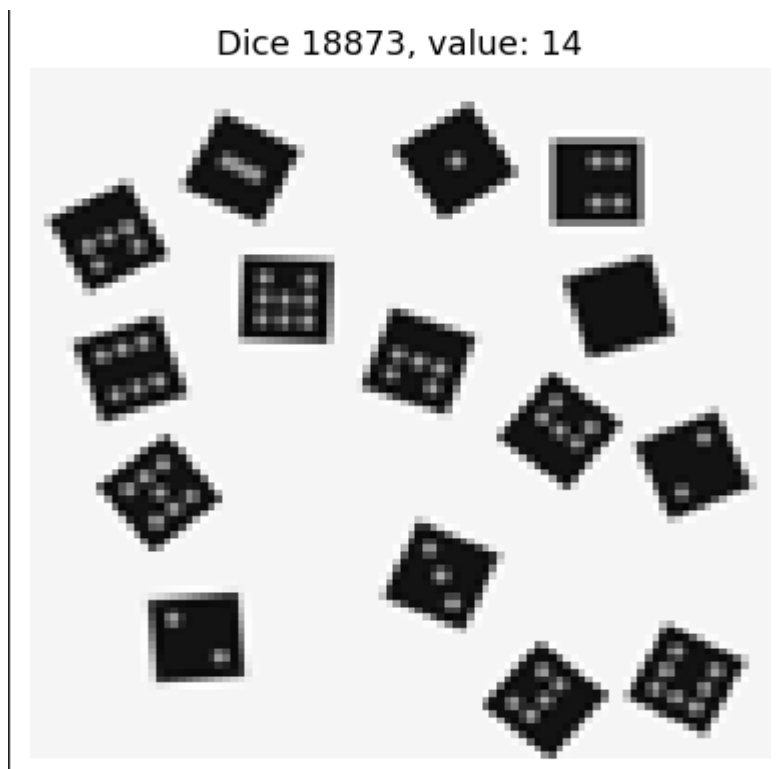
```

Wynik weryfikacji:

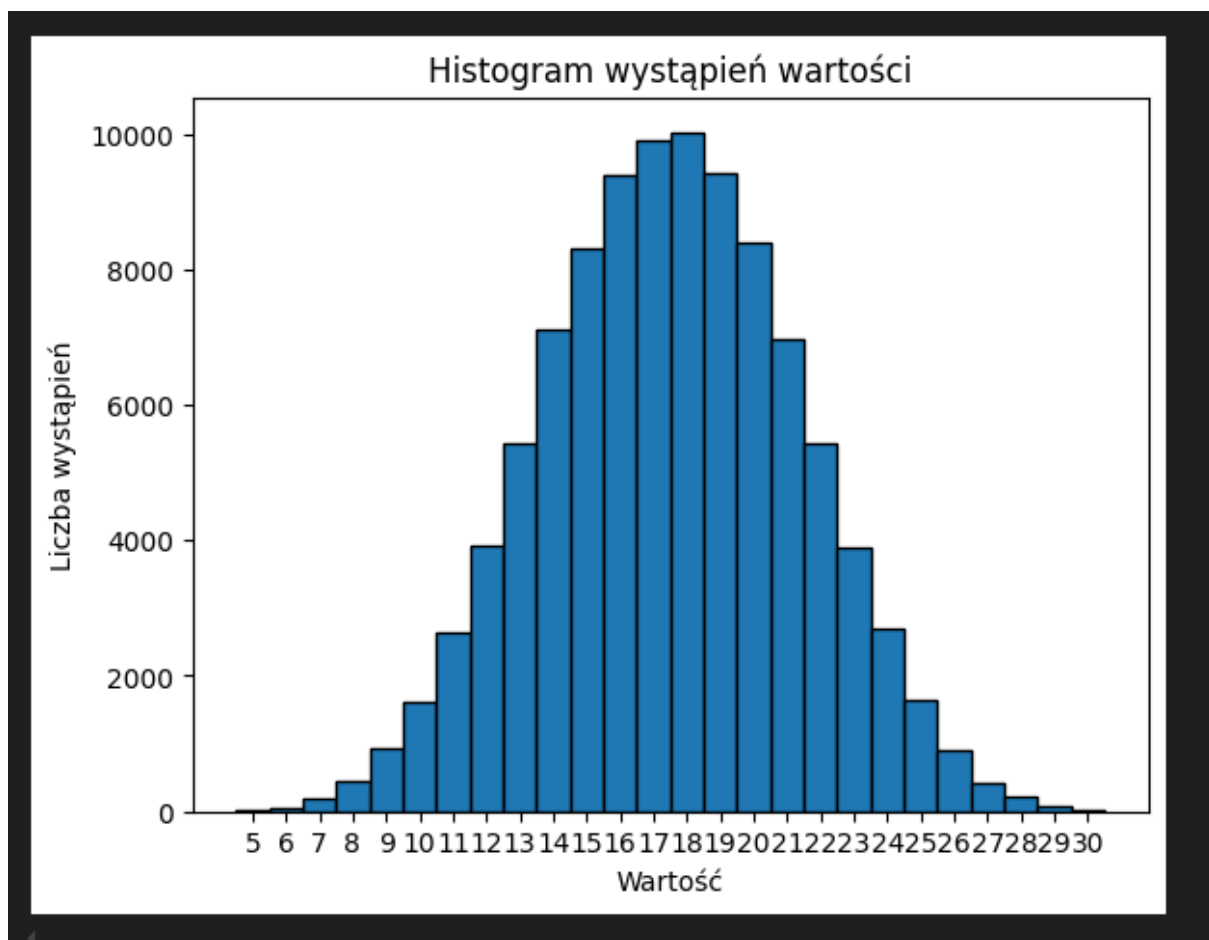
```
dice:
Label Shape: (100000,) has NAN: False
Values : [ 5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28
 29 30]
Data Shape: (100000, 100, 100) has NAN: False
Values : [  0   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17
 18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35
 36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53
 54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71
 72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89
 90  91  92  93  94  95  96  97  98  99 100 101 102 103 104 105 106 107
108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125
126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143
144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161
162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179
180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197
198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215
216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233
234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251
252 253 254 255]
```

Weryfikacji pokazała nie zgodność rzeczywistych danych z wyżej wymienionymi założeniami jakie powinny spełniać, ponieważ rzeczywiste etykiety są z zakresu <5, 30>, a założenie było <6, 30>, wartości pikseli są zgodne z założeniami.

Przykładowy wyświetlony obraz zgodny z założeniem (nr = index modulo liczba obrazów):



Sprawdziłem dodatkowo licznosc wystąpień i utworzyłem z nich histogram:



Jak się spodziewałem większość wartości jest średnią z minimum i maksimum $(5 + 30) / 2 = 17.5$, zatem najliczniejsze klasy to suma wynosząca 17 i 18 oczek.