به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

Trustworthy AI

تمرین شماره ۲

نام و نام خانوادگی	محمد جواد رنجبر
شماره دانشجویی	۸۱۰۱۰۱۱۷۳
تاریخ ارسال گزارش	

فهرست گزارش سوالات

سوال ۱ – SHAP

الف)

۱. یک روش additive feature attribution می‌توان به deeplift اشاره کرد. در این روش که مختص مدل‌های عمیق می‌باشد. به هر ورودی x_i یک مقدار $C_{\Delta x_i \Delta y}$ نسبت می‌دهد که نشان‌دهنده اثر تنظیم شدن ورودی به یک مقدار مرجع بر خلاف مقدار اصلی آن است. این به این معنی است که برای DeepLIFT، نگاشت $x = h(x')$ مقادیر باینری را به ورودی‌های اصلی تبدیل می‌کند، جایی که ۱ نشان می دهد که یک ورودی مقدار اصلی خود را می‌گیرد و ۰ نشان می دهد که مقدار مرجع را می‌گیرد. مقدار مرجع، اگرچه توسط کاربر انتخاب شده است، یک مقدار پس‌زمینه معمولی غیر اطلاعاتی را برای این ویژگی نشان می‌دهد. DeepLIFT از ویژگی "summation-to-delta" استفاده می‌کند که بیان می‌کند:

$$\sum_{1=1}^{n} C_{\Delta x_i \Delta o} = \Delta o$$

که در آن $o = f(x)$ خروجی مدل است، $\Delta x_i = x_i - r_i$، $\Delta o = f(x) - f(r)$ و r ورودی مرجع است. که در صورتی که $\varphi_i = C_{\Delta x_i \Delta o}$ و $\varphi_i = f(r)$ باشد، معادله additive feature attribution برای این روش برقرار است.

Local accuracy:

به معنی این است که پیش‌بینی مدل اصلی یا f برای ورودی x، حداقل معادل پیش‌بینی مدل explanation یا g برای ورودی ساده‌سازی شده x’ باشد. یعنی برای $x = h_x(x')$ پیش‌بینی $g(x')$ معادل $f(x)$ باشد و به صورت زیر می‌توان بیان کرد:

$$f(x) = g(x') = \varphi_O + \sum_{i=1}^{M} \varphi_i x'$$

Missingness:

این صفت به این معنی است که ویژگی که در ورودی اصلی وجود ندارد هیچ تاثییری نداشته باشد. به عبارت دیگر صفت missingness ورودی $x' = 0$ را مجبور می‌کند که اثری نداشته باشد.

$$sx' = 0 \Rightarrow \varphi_i = 0$$

Consistency:

در صورتی که مدل به گونه‌ای تغییر کند که مشارکت بعضی از ورودی‌های ساده شده جدا از سایر ورودی‌ها تغییر کند، تاثییر ورودی نباید عوض شود. برای $f_x(z') = f(h_x(z'))\ and\ z'_i = 0$ و دو مدل f و f' داریم:

$$f_x(z') - f_x(z'\backslash i) \geq f_x(z') - f_x(z'\backslash i)$$

و برای تمام ورودی‌های $z'\epsilon\{0,1\}^m$ داریم: $\varphi_i(f', x) \geq \varphi_i(f, x)$

۲. *Kernel SHAP* یک روش آگنوستیک مدل برای تقریب مقادیر *SHAP* با استفاده از ایده‌هایی از مقادیر *LIME* و *Shapley* است. در مقاله *Shap*، نویسندگان نشان می‌دهند که با یک مدل رگرسیون خطی وزن‌دار به عنوان مدل جایگزین محلی و یک *kernel* مناسب، ضرایب رگرسیون مدل جایگزین *LIME* مقادیر *SHAP* را تخمین می‌زند. *Shapley kerel* که مقادیر *SHAP* را بازیابی می کند توسط معادله زیر پیدا می‌شود:

$$\pi_{x'}(z') = \frac{(M\ -\ 1)}{(M\ choose\ |z'|)|z'|(M\ -\ |z'|)}$$

جایی که M تعداد ویژگی ها و $|z'|$ است تعداد ویژگی های غیر صفر در ورودی ساده شده z' است.

۳. در *deepshap* از ایده‌ی *deeplift* الگو گرفته‌اند و *Deep SHAP* مقادیر *SHAP* محاسبه شده برای اجزای کوچکتر شبکه را در مقادیر *SHAP* برای کل شبکه ترکیب می‌کند. برای مدل‌های یادگیری عمیق مانند شبکه‌های عصبی طراحی شده است. این یک توسعه از روش *SHAP* است که برای این نوع مدل‌ها بهینه شده است.

Deep SHAP از نظر محاسباتی کارآمدتر از *Kernel SHAP* است که با مدل‌های یادگیری عمیق سروکار دارد، زیرا می‌تواند از ساختار مدل و انتشار پس‌انداز برای محاسبه مؤثر مقادیر *Shapley* استفاده کند. این به ویژه برای توضیح پیش‌بینی‌های تصویر، متن و داده‌های صوتی پردازش شده توسط مدل‌های یادگیری عمیق مفید است.

Kernel SHAP یک روش مدل-آگنوستیک است، به این معنی که می‌توان آن را برای هر نوع مدلی از جمله مدل‌های خطی، درخت تصمیم و مدل‌های یادگیری عمیق اعمال کرد. از یک تابع *kernel* طراحی شده ویژه برای تقریب مقادیر *Shapley* بر اساس تعداد محدودی نمونه از فضای ورودی استفاده می‌کند. می‌تواند از نظر محاسباتی گران باشد، به ویژه برای فضاهای ورودی با ابعاد بالا یا مدل‌های پیچیده، زیرا نیاز به نمونه‌برداری و ارزیابی مجدد مدل برای ترکیب‌های ورودی مختلف دارد. نسبت به *Deep SHAP*

انعطاف‌پذیرتر است، زیرا می‌توان آن را برای طیف گسترده‌ای از انواع مدل‌ها اعمال کرد، اما ممکن است از نظر محاسباتی برای مدل‌های یادگیری عمیق کارآمد نباشد.

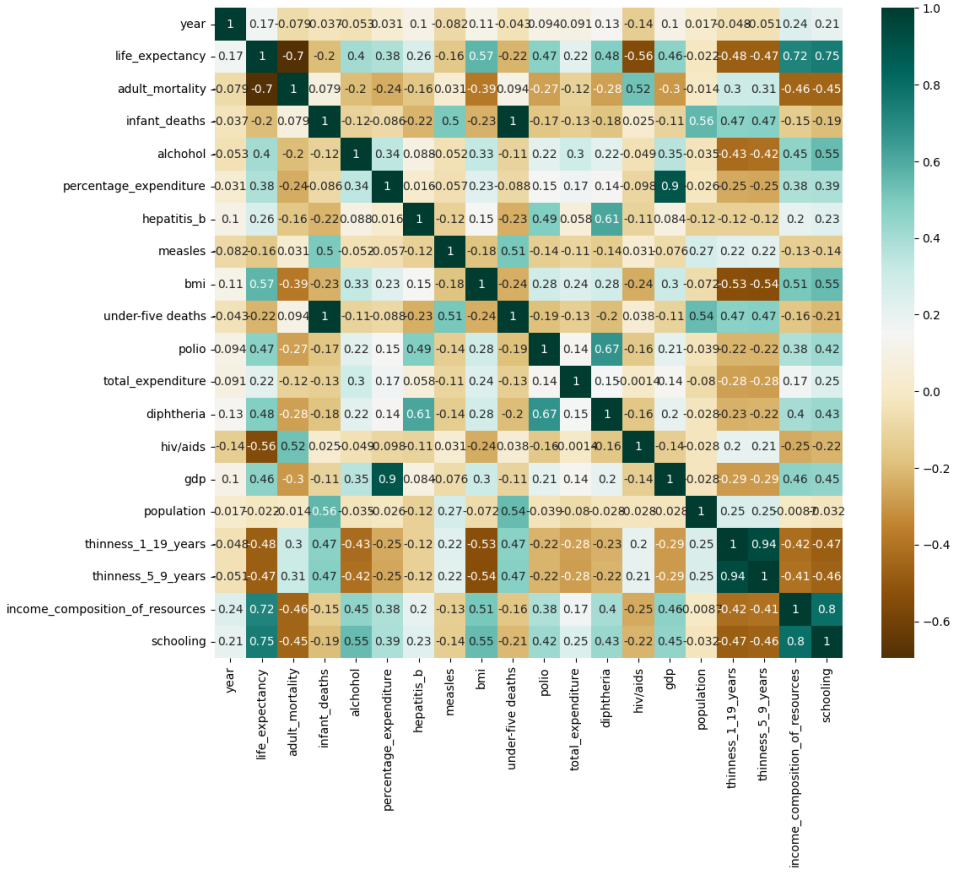
ب)

ابتدا داده‌ها را لود می‌کنیم و دیتاست را برای *missing value*ها بررسی می‌کنیم.

```
Country                             0
Year                                0
Status                              0
Life expectancy                    10
Adult Mortality                    10
infant deaths                       0
Alcohol                           194
percentage expenditure              0
Hepatitis B                       553
Measles                             0
 BMI                               34
under-five deaths                   0
Polio                              19
Total expenditure                 226
Diphtheria                         19
 HIV/AIDS                           0
GDP                               448
Population                        652
 thinness  1-19 years              34
 thinness 5-9 years                34
Income composition of resources   167
Schooling                         163
dtype: int64
```

شکل ۱ **missing value**های مجموعه داده

همچنین برای درک بهتر داده‌ها می‌توانیم کورولیشن ماتریس ویژگی‌ها را رسم کنیم که شکل زیر را خواهد داشت:



شکل ۲ کورولیشن ویژگی‌ها

حال ویژگی‌های missing و نادرست را باید جایگزین کنیم. سیاست جایگزینی ویژگی‌های مختلف می‌کند برای مثال تعدادی از ویژگی‌ها وابسته به *developed* یا *developing* بودن کشورها میانگین این ویژگی‌ها را جایگزین می‌کنیم برای تعدادی میانگین کلی یا مقادیر سال‌های گذشته را جایگزین می‌کنیم.

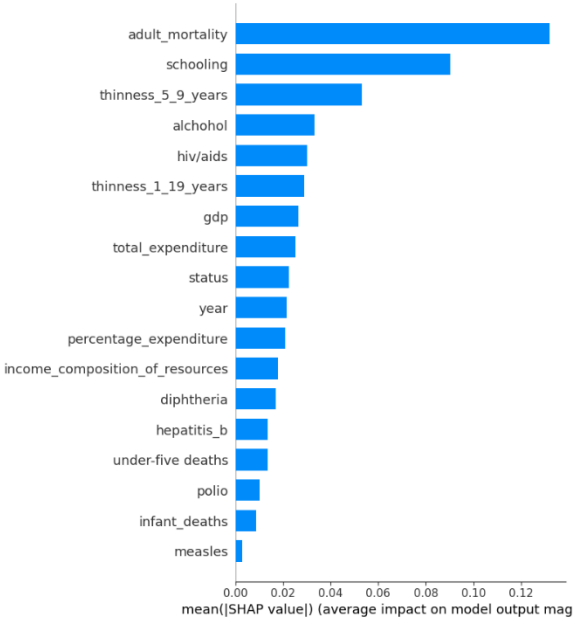
بعد از تمیز کردن ویژگی‌ها، این ویژگی‌ها را نورمالایز می‌کنیم تا مدل راحتتر آموزش ببیند. همچنین

ویژگی‌های categorical را به عددی تبدیل می‌کنیم. مدل طراحی شده به شکل زیر می‌باشد:

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 128)               2432

 dense_1 (Dense)             (None, 64)                8256

 dense_2 (Dense)             (None, 1)                 65

=================================================================
Total params: 10,753
Trainable params: 10,753
Non-trainable params: 0
_________________________________________________________________
```

شکل ۳ معماری مدل رگرشن

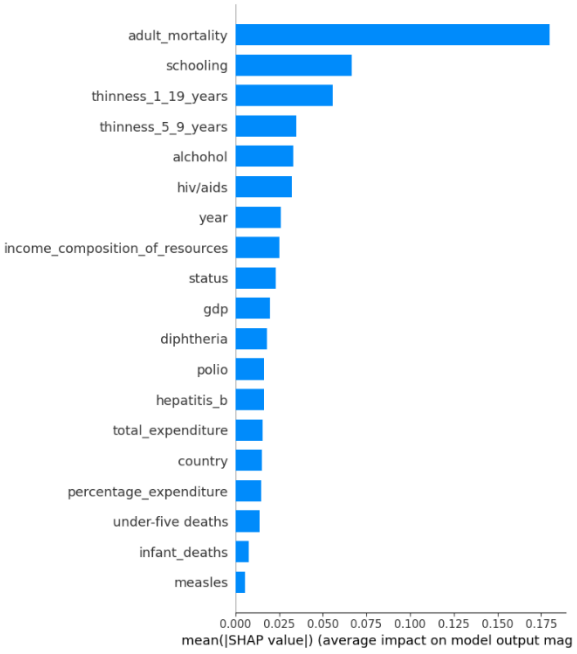
حال مدل را آموزش می‌دهیم و برای دو حالت deepshap و kernelshap نتایج به صورت زیر می‌باشد:



شکل ۴ ویژگی‌ها با مدل **deepshap**

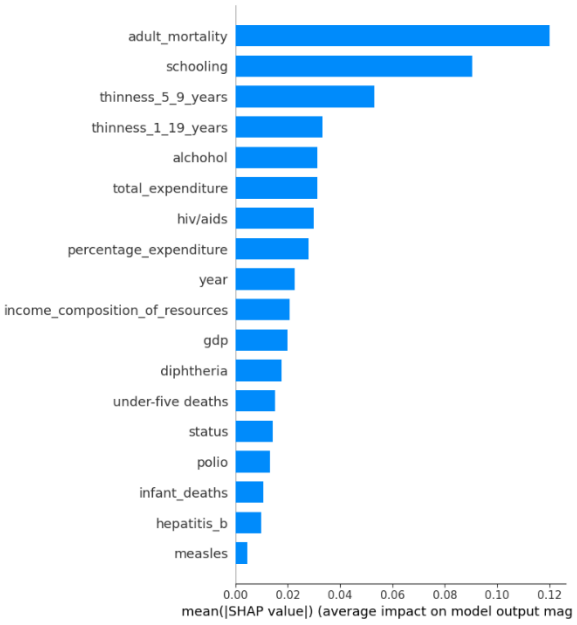
Adult mortality به تعداد مرگ و میر افراد ۱۵ تا ۶۰ ساله در یک جمعیت اشاره دارد. این شاخص مهمی از سلامت و رفاه کلی یک جمعیت است، زیرا نشان دهنده شیوع بیماری ها، دسترسی به مراقبت های بهداشتی و سایر عوامل اجتماعی-اقتصادی است. از سوی دیگر، امید به زندگی میانگین سال‌هایی است که انتظار می‌رود یک فرد معمولاً از بدو تولد زندگی کند. مشخص است که این ویژگی تاثییر زیادی بر روی Life expectancy خواهد داشت. همچنین پیش از استفاده از مدل نیز این ویژگی با Life expectancy کوررولیشن بالایی داشت. ویژگی‌های بعدی شامل schooling و سایر ویژگی‌های مرتبط به سلامت به ترتیب در تصمیم‌گیری مدل تاثییر دارند.

ویژگی کشور را حذف کردیم چون معنی خاصی نداشت با این حال با در نظر گرفتن این ویژگی نمودار به شکل زیر می‌شود.



شکل ۵ **deepshap** با کشور

تصویر به دست آمده از kerenlshap نیز تقریبا همین شکلی می‌باشد:



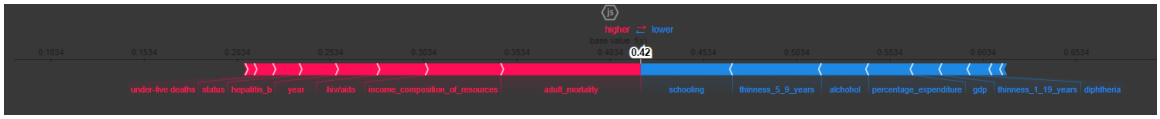
شکل ۶ ویژگی‌ها با مدل **kernelshap**

Forceplote نمایش‌دهنده این است که کدام ویژگی بیشترین تأثیر را در پیش بینی مدل برای یک مشاهده واحد داشته است. حال forceplot را برای دو کشور سوریه و مالزی رسم می‌کنیم:

شکل ۷ **forceplot** کشور سوریه

شکل ۸ **forceplot** مالزی

نمودار براش روش کرنل نیز تقریبا به همین شکل خواهد بود:



شکل ۹ **forceplot** سوریه

شکل ۱۰ **forceplot** مالزی

حال با توجه به نتایج این دو کشور ویژگی‌های مختلفی اهمیت دارد برای مثال مشخص است که life expediency مالزی بسیار بیشتر از سوریه است، ویژگی‌های قرمز، آن‌هایی هستند که باعث افزایش این امتیاز و ویژگی‌های آبی آن‌هایی هستند که باعث کاهش این امتیاز شده‌اند. در این دو کشور ویژگی‌های مختلفی تاثییر مثبت یا منفی داشته‌اند.

سوال ۲ – Knowledge Distillation

۱- شبکه‌های عصبی به شدت در وظایفی که به آن‌ها داده می‌شود خوب عمل می‌کنند. اما با وجود تعداد لایه‌های زیاد، رابطه بین ورودی و خروجی در این شبکه‌ها بسیار پیچیده است و دلیل تصمیم‌گیری‌های آن‌ها مشخص نیست. درخت‌های تصمیم‌گیری نرم[۱] از شبکه های عصبی قابل تفسیر هستند و می توانند برای یادگیری انتقالی[۲] استفاده شوند. آنها همچنین پارامترهای کمتری نسبت به شبکه های عصبی دارند و می توان آنها را سریعتر آموزش داد.

[۱] Soft decision tree
[۲] Transfer learning

۲- مدل معرفی شده به جای استفاده از سلسله مراتب ویژگی اطلاعاتی که از شبکه‌ی عصبی به دست آمده برای آموزش درخت تصمیم استفاده می‌شود. دانش در مدلی که به جای ویژگی‌ها بر تصمیمات سلسله مراتبی متکی است، توضیح یک تصمیم خاص بسیار آسان‌تر خواهد بود.

٤- تابع هزینه استفاده شده در مدل آنها مجموع دو عبارت است: آنتروپی متقاطع بین اهداف نرم تولید شده توسط شبکه معلم و احتمالات تولید شده توسط شبکه دانش آموز، و آنتروپی احتمالات تولید شده توسط شبکه دانش آموزی. اصطلاح آنتروپی متقاطع، شبکه دانش آموز را تشویق می کند تا احتمالاتی را تولید کند که نزدیک به احتمالات تولید شده توسط شبکه معلم است. اصطلاح آنتروپی شبکه دانش‌آموزی را تشویق می‌کند تا احتمالاتی را تولید کند که تا حد امکان نزدیک به یکنواخت باشد. این تأثیری دارد که شبکه دانش‌آموزی را تشویق می‌کند تا تصمیم‌های «نرم» اتخاذ کند که اعتماد به نفس کمتری نسبت به تصمیم‌گیری‌های شبکه عصبی استاندارد دارند.
این تابع هزینه که به دنبال به حداقل رساندن آنتروپی متقاطع بین هر برگ، وزن‌دار شده بر اساس احتمال مسیر آن، و توزیع هدف است. برای یک مورد آموزشی با بردار ورودی x و توزیع هدف T، این تابع هزینه به شکل زیر است:

$$L(\mathrm{x}) = -\log\Big(\sum_{\mathrm{l}\epsilon\ \mathrm{Leaf\ Nodes}} P^{\mathrm{l}}(\mathrm{x}) \sum_{k} T_k \log Q_k^{\mathrm{l}}\Big)$$

جایی که T توزیع هدف و $P^{\mathrm{l}}(\mathrm{x})$ احتمال رسیدن به گره l برگ با توجه به ورودی x است.

٥- برای جلوگیری از گیر افتادن در راه حل‌های ضعیف در طول آموزش، یک ترم جریمه یا معرفی شده است که هر گره داخلی را تشویق می کرد تا از هر دو زیر درخت چپ و راست استفاده مساوی کند. بدون این regularization ترم، درخت تمایل داشت در لوکال مینیم‌هایی گیر کند که در آن یک یا چند گره داخلی همیشه تقریباً تمام احتمالات را به یکی از درختان فرعی آن اختصاص می دادند (راست یا چپ) و گرادیان لجستیک برای این تصمیم همیشه بسیار نزدیک به صفر بود.

سوال ۳ – D-rise

(a

برای توضیح عملکرد مدل در روش‌های پیشین فقط به پیش‌بینی مدل کفایت می‌کردند اما در مدل D-rise علاوه بر پیش‌بینی مدل محل این پیش‌بینی و bounding box اطراف آن نیز اهمیت دارد.

همچنین در روش‌های قدیمی بیشتر روی وظیفه Image classification تمرکز شده بود، اما در این روش بر روی وظایف دیگری نیز تمرکز می‌شود.

روش D-rise از روش ماسک کردن rise الگو گرفته است و بدون نیاز به دانستن معماری داخلی مدل یا گرادیان ورودی می‌تواند روی هر object detector اعمال شود.

b) این روش به صورت زیر است:

۱. N نمونه ماسک باینری با اندازه $h \times w$ (کوچکتر از اندازه تصویر $H \times W$) انتخاب می‌کنیم. این نمونه‌ها با احتمال p مستقل از سایر نمونه‌ها برابر با یک و با احتمال 1-p برابر با صفر خواهند بود.

۲. نمونه برداری از همه ماسک ها به اندازه $(h+1)C_H \times (w+1)C_W$ با استفاده از درونیابی دوخطی، که در آن $C_W{}^*C_H = \left\lfloor \frac{H}{h} \right\rfloor * \left\lfloor \frac{W}{w} \right\rfloor$ اندازه سلول در ماسک نمونه‌برداری شده است.

۳. مناطق $H \times W$ را با احتمال تصادفی یکنواخت از (۰،۰) تا (C_W, C_H) کراپ می‌کنیم.

c) معیارهای شباهت این مقاله به صورت زیر می‌باشند.

برای پیدا کردن شباهت bounding box از معیار Intersection over Union (IoU) استفاده شده است. برای پیدا کردن شباهت ناحیه‌ها از معیار cosine similarity استفاده شده است. همچنین برای مدل‌هایی مانند YOLOv3 که objectness score فراهم می‌کنند که برای قرار دادن این در معیار شباهت O_j را در این معیار ضرب می‌کنند. ناحیه‌هایی با احتمال وجود شی پایینتر شباهت کمتری به بردار هدف دارند. همچنین برای مدل‌هایی که objectness score فراهم نمی‌کنند از این ترم صرف نظر می‌شود. به طور کلی به صورت ریاضی این معیار به شکل زیر خواهد بود:

$$s(d_t, d_j) = s_L(d_t, d_j)^* s_P(d_t, d_j)^* s_O(d_t, d_j)$$

که معیارهای ما به صورت زیر می‌باشند:

$$s_L(d_t, d_j) = IoU(L_t, L_j)$$

$$s_P(d_t, d_j) = \frac{P_t + P_j}{\|P_t\| \|P_j\|}$$

$$s_O(d_t, d_j) = O_j$$

که شباهت حاصل از and این سه معیار می‌باشد و در صورتی که یکی کم باشد حاصل نهایی نیز کم خواهد بود.

d) برای بهبود مقاله به موارد زیر می‌توان اشاره کرد:

e) تصویر اول این تصویر از ساندویچ لیوان و میزغذا خوری است:

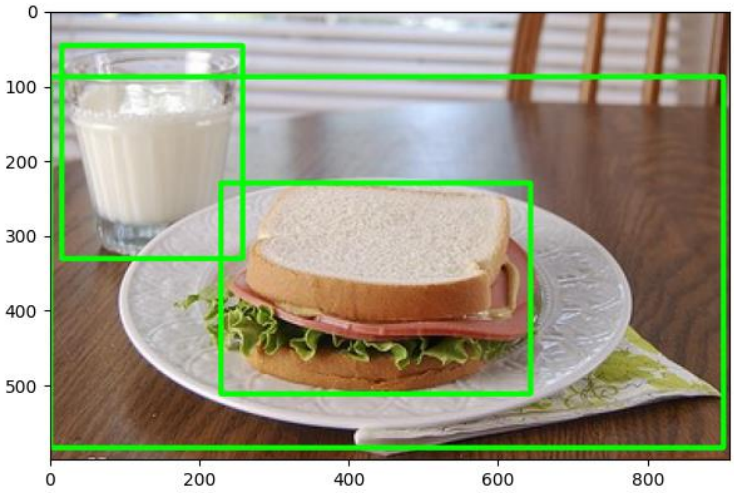
Figure ۱ ساندویچ، لیوان و میز غذاخوری

پیش‌بینی مدل و مکان boundingboxهای این عکس به شرح زیر می‌باشد:

```
41 cup (15, 46, 257, 331) 0.9972319
48 sandwich (228, 230, 642, 512) 0.97169656
60 dining table (0, 88, 900, 584) 0.9807834
```

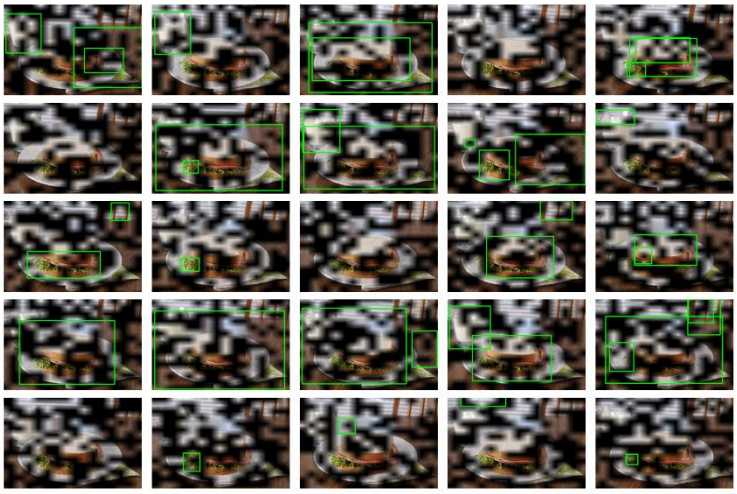
شکل ۱۱ پیش‌بینی مدل برای عکس ساندویچ

که این پیش‌بینی را در تصویر زیر مشاهده می‌کنید:



شکل ۱۲ نمایش پیش‌بینی مدل برای عکس ساندویچ

همچنین ماسک شده تصویر به صورت زیر خواهد بود:



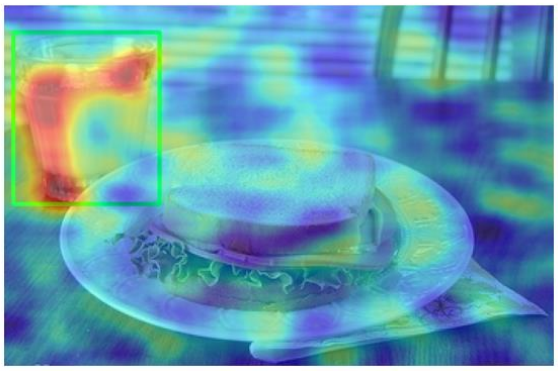
شکل ۱۳ ماسک شده تصویر ساندویچ

Saliency map مدل برای ساندویچ به شکل زیر می‌باشد:

با توجه به تصویر مشخص است که بیشتر توجه تصویر بر روی خود ساندویچ بوده، با این حال اطراف ساندویچ نیز مقداری مورد توجه مدل بوده، این اتفاق می‌تواند به دلیل context aware بودن مدل‌ها باشد که علاوه بر خود ساندویچ به اطراف آن در عکس نیز توجه می‌کنند و باعث بهتر شدن تشخیص ساندویچ می‌شوند.

تصویر Saliency map برای لیوان به شکل زیر می‌باشد:

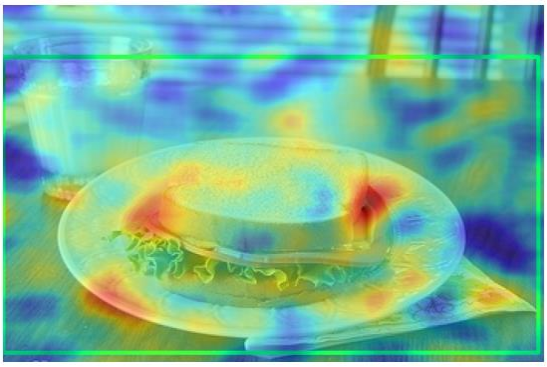
توضیحات بالا راجع به تشخیص مدل برای لیوان نیز صدق می‌کند.



شکل ۱۴ **Saliency map** برای لیوان

تصویر Saliency map برای میز غذاخوری به شکل زیر می‌باشد:

برای میز غذاخوری نیز بخش زیادی از توجه به ساندویچ هست که باز به دلیل context aware بودن مدل با توجه به موارد روی میز خود میز را تشخیص داده است.



شکل ۱۵ **Saliency map** میز غذاخوری

عکس دوم شامل یک پرنده و یک تخته موج‌سواری می‌باشد:

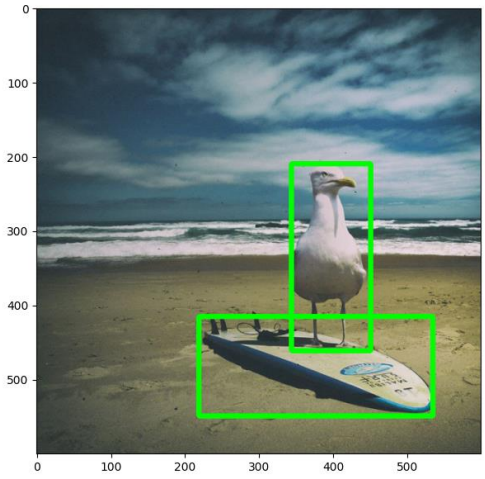
شکل ۱۶ پرنده و تخته موج‌سواری

پیش‌بینی مدل برای این تصویر به شرح زیر است:

```
14 bird (344, 209, 451, 461) 0.9977143
37 surfboard (219, 415, 535, 549) 0.9345766
```

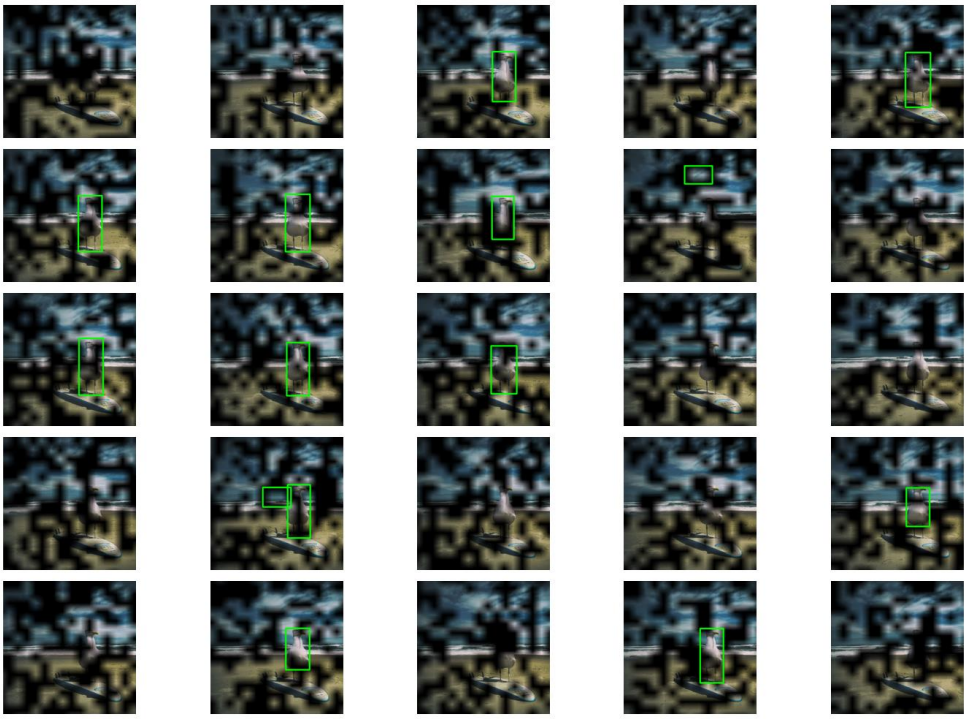
شکل ۱۷ پیش‌بینی مدل برای تصویر تخته موج‌سواری

که این پیش‌بینی را در تصویر زیر مشاهده می‌کنید:



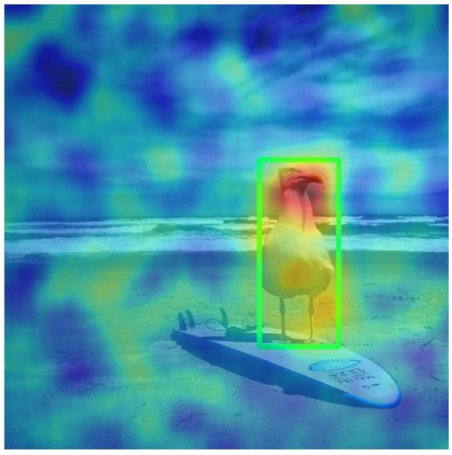
شکل ۱۸ نمایش پیش‌بینی مدل برای تصویر تخته موج‌سواری

همچنین ماسک شده تصویر به صورت زیر خواهد بود:



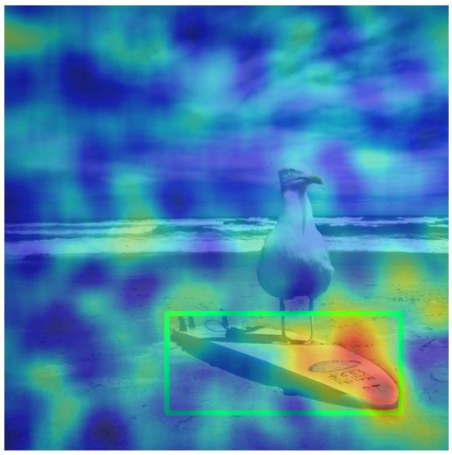
شکل ۱۹ ماسک شده تصویر تخته موج‌سواری

تصویر Saliency map برای پرنده به شکل زیر می‌باشد:



شکل ۲۰ **Saliency map** تصویر پرنده

Saliency map برای تصویر تخته موج‌سواری به شکل زیر می‌باشد:



شکل ۲۱ **Saliency map** برای تخته موج‌سواری

برای اشیا این تصویر نیز بیشتر توجه مدل به خود شی بوده، اما برای تخته موج‌سواری اطراف و شن ساحل نیز به نظر مورد توجه مدل قرار گرفته است.

تصویر سوم شامل یک اسب و چتر می‌باشد:

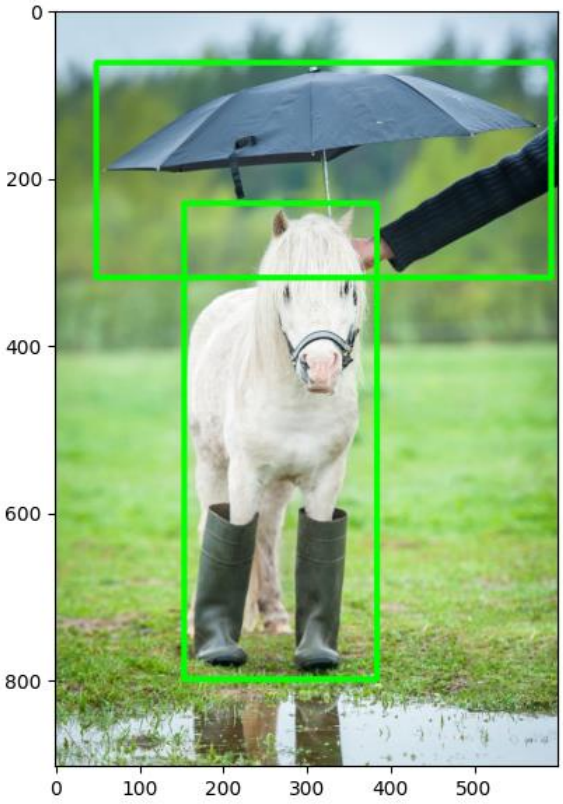
شکل ۲۲ اسب و چتر

پیش‌بینی مدل برای این تصویر به شرح زیر می‌باشد:

```
17 horse (154, 229, 384, 798) 0.9968617
25 umbrella (49, 61, 592, 318) 0.94581425
```

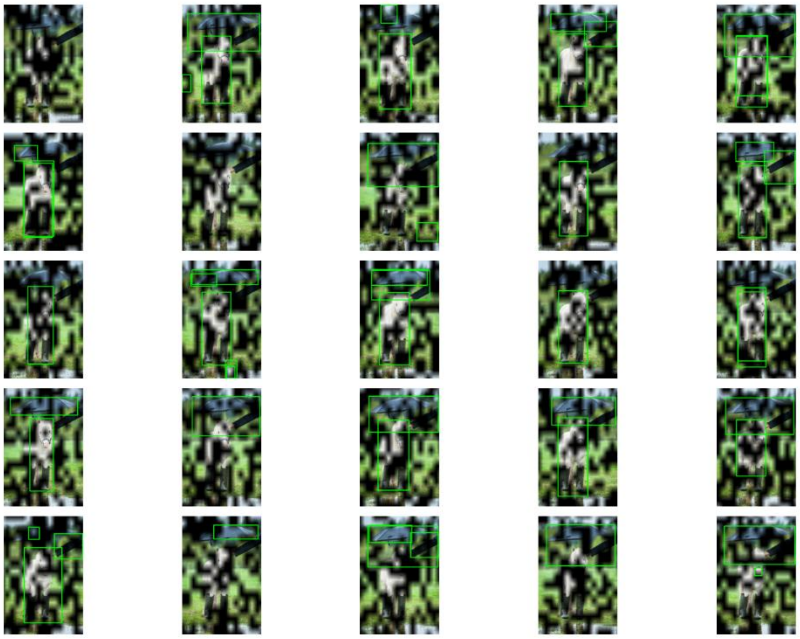
شکل ۲۳ پیش‌بینی مدل برای تصویر اسب

که این پیش‌بینی را در تصویر زیر مشاهده می‌کنید:



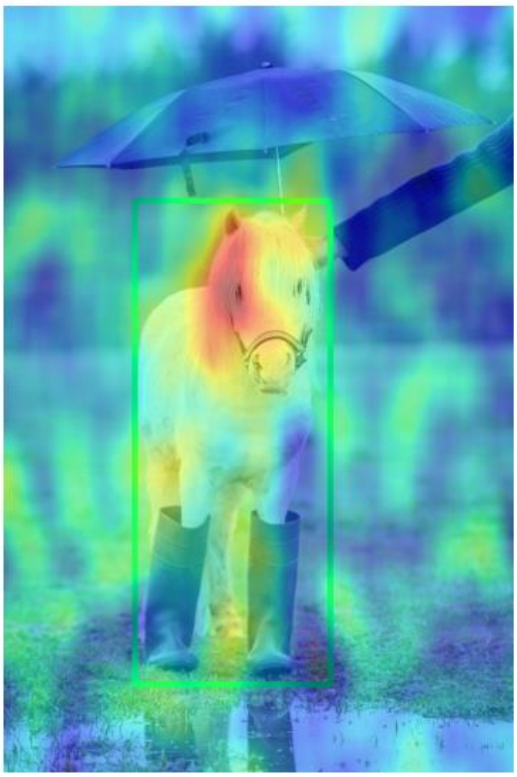
شکل ۲۴ نمایش پیش‌بینی مدل برای تصویر اسب

همچنین ماسک شده تصویر به صورت زیر خواهد بود:



شکل ۲۵ ماسک شده تصویر اسب

Saliency map برای تصویر اسب به شکل زیر می‌باشد:



شکل ۲۶ **Saliency map** برای تصویر اسب

که مدل بیشتر به صورت اسب توجه کرده و منطقی است و مثلا چتر که ربطی به اسب ندارد در پیدا کردن اسب هیچ تاثییری ندارد.

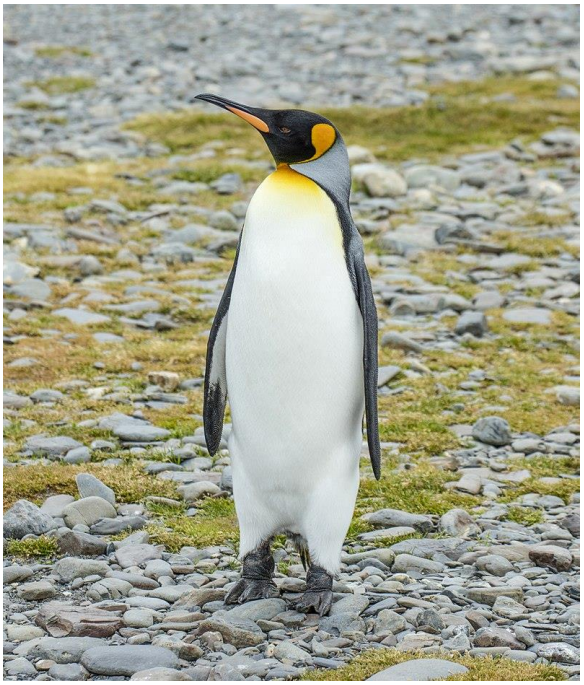
Saliency map برای تصویر چتر به شکل زیر می‌باشد:

شکل ۲۷ **Saliency map** برای تصویر چتر

در این تصویر نیز مدل بیشتر به چتر و زمین اطراف توجه کرده و اسب تقریبا بی‌تاثییر است.

سوال ۴ – LIME

ابتدا مدل را load کرده و تصویر اول یک عکس از king penguin انتخاب شده است که پیش‌بینی مدل به شرح زیر است.



شکل ۲۸ تصویر **king penguin**

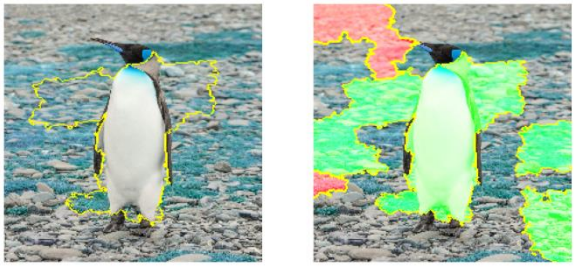
که پیش‌بینی‌های مدل برای این تصویر به صورت زیر است:

```
('n02056570', 'king_penguin', 0.80619997)
('n02071294', 'killer_whale', 0.009418877)
('n03950228', 'pitcher', 0.008754731)
('n01855032', 'red-breasted_merganser', 0.00556122)
('n02074367', 'dugong', 0.0050408235)
```

شکل ۲۹ پیش‌بینی مدل برای تصویر پنگوئن

که بیشترین پیش‌بینی مربوط به king penguin می‌باشد.

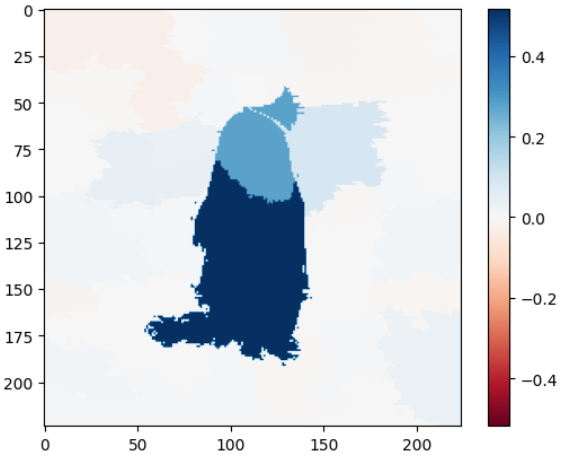
حال pros and cons و boundary این تصویر به شکل زیر می‌باشد:



شکل ۳۰ **pros and cons** و **boundary**های تصویر پنکوئن

مشخص است که پنگوئن و بدنش به عنوان مهم‌ترین بخش‌ها برای این تصمیم انتخاب شده‌اند اما نواحی اطراف مانند سنگ‌ها کمترین اهمیت را داشته‌اند.

حال heatmap این تصویر نیز به شکل زیر می‌باشد:



شکل ۳۱ **heatmap** تصویر پنگوئن

در تصویر بالا نیز اهمیت بدن پنگوئن برای این تصمیم نشان داده شده است.

تصویر دوم شامل دو کلاس مختلف گربه و سگ است:

شکل ۳۲ تصویر گربه و سگ

که پیش‌بینی‌های مدل برای این تصویر به صورت زیر است:

```
23045', 'tabby', 0.05190673)
23159', 'tiger_cat', 0.036255352)
12137', 'chow', 0.032276053)
93428', 'American_Staffordshire_terrier', 0.032
24075', 'Egyptian_cat', 0.03149184)
```

شکل ۳۳ پیش‌بینی مدل برای تصویر گربه و سگ

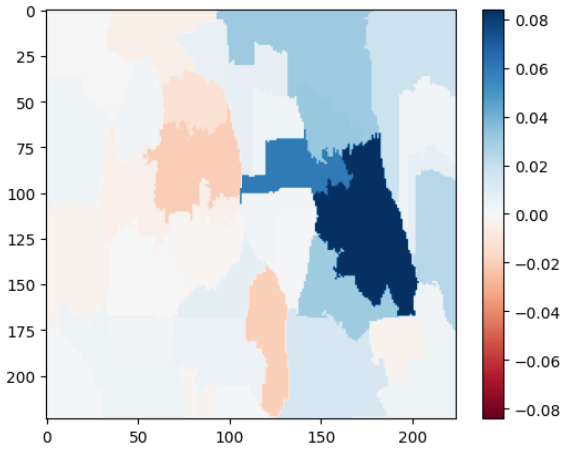
تمام ۵ پیش‌بینی اول این مدل شامل نژادهای مختلف سگ و گربه است که منطقی است.

حال pros and cons و boundary این تصویر به شکل زیر می‌باشد:

شکل ۳۴ **pros and cons** و **boundary**های تصویر گربه و سگ

از آنجا که مدل گربه را به عنوان پیش‌بینی اول این تصویر تشخیص داده است، مهم‌ترین بخش‌های این عکس شامل گربه بوده و سگ تاثیر منفی در این پیش‌بینی داشته است.

heatmap این تصویر نیز به شکل زیر می‌باشد:



شکل ۳۵ **heatmap** تصویر گربه و سگ

در heatmap نیز مشخص است که بدن گربه به عنوان بخش مهم تصویر انتخاب شده است.

تصویر سوم نیز شامل یک عکس از یک نوع میمون به نام شست بریده است.

شکل ۳۶ تصویر میمون شست بریده

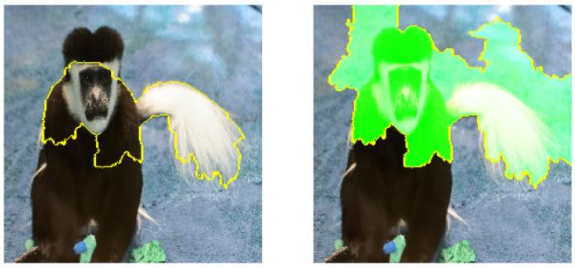
که پیش‌بینی‌های مدل برای این تصویر به صورت زیر است:

```
('n02488702', 'colobus', 0.8087967)
('n02484975', 'guenon', 0.04300543)
('n02483362', 'gibbon', 0.011951433)
('n02488291', 'langur', 0.0063018017)
('n02493509', 'titi', 0.005029436)
```

شکل ۳۷ پیش‌بینی مدل برای میمون

این پیش‌بینی‌ها شامل نژادهای مختلف میمون می‌باشد که میمون شست بریده بیشترین دقت را دارا است.

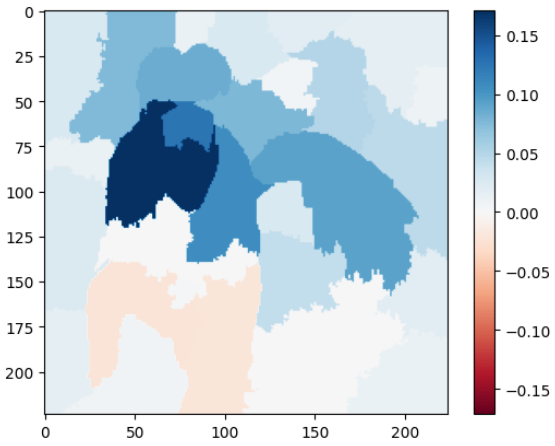
حال pros and cons و boundary این تصویر به شکل زیر می‌باشد:



شکل ۳۸ **pros and cons** و **boundary**های تصویر میمون

بدن و صورت میمون به عنوان بخش مهم تصویر انتخاب شده است.

حال heatmap این تصویر نیز به شکل زیر می‌باشد:



شکل ۳۹ **heatmap** برای تصویر میمون

در heatmap نیز اهمیت بدن و صورت میمون نشان داده شده است.