Politechnika Wrocławska

Podstawy Optymalizacji

Katedra Automatyki, Mechatroniki i Systemów Sterowania

Problem Regresji Liniowej Spadek gradientowy

Marcin Nurzyński
226 232

Marcin Woźniak
226 399

24 maja 2018

Spis treści

1 Wprowadzenie

Tematem projektu jest pokazanie na przykładzie cen mieszkań w jaki sposób, używając zbioru danych, możemy przewidywać, jakie wartości będą miały kolejne pary danych. Technika używana przez nas do rozwiązania tego problemu to uzyskanie regresji liniowej przy użyciu spadku gradientowego. Cały problem składa się z dwóch części - teoretycznej oraz praktycznej.

1.1 Część teoretyczna

W części teoretycznej zostaną wytłumaczone wszystkie kroki na przykładowych obliczeniach, które są niezbędne do wyznaczenia prostej, która w najdokładniejszy sposób opisze zbiór danych. Dzięki tej prostej będziemy w stanie ocenić najbardziej prawdopodobną daną wyjściową dla pewnym danych wejściowych. Do uzyskania tego policzymy funkcję kosztu dla dwóch możliwych prostych, które będą w pewien sposób odwzorowywały te dane. Dzieki temu zrozumiemy na czym polega funkcja kosztu i dlaczego jej pochodna jest nam potrzebna do spadku gradientowego, który odnajdzie najlepsze θ_0 oraz θ_1.

1.2 Część praktyczna

Praktyczna część będzie polegała na stworzenie programu, który sam będzie obliczał funkcję kosztu oraz stworzy z tego wykres kosztu, na którym potem przeprowadzi spadek gradientowy. Dzięki temu zabiegowi otrzymamy predykcję cen mieszkań w zależności od metrażu. Całe oprogramowanie zostanie napisane w Octave - darmowym odpowiednikiem Matlab'a. To środowisko programistyczne zostało wybrane ze względu na prosty, a przede wszystkim czytelny sposób przeprowadzania obliczeń. Każda część będzie odwzorowana za pomocą wykresów, które będą również przedstawiały kroki, tak aby jak najlepiej zoobrazować i wytłumaczyć w połączeniu z częścią teoretyczną ideę tego problemu. Spadek gradientowy dla tak błachego problemu jest przerostem formy nad treścią, aczkolwiek chodzi tutaj o jak najlepsze wytłumaczenie techniki.

2 Opis problemu

2.1 Cel algorytmu

Celem algorytmu jest przedstawienie regresji liniowej i jednego ze sposób, związengo z popularnym aktualnie uczeniem maszynowym, radzenia sobe z obliczeniem jej - spadku gradientowego. Przedstawiony tutaj algorytm będzie przewidywał cenę mieszkań we Wrocławiu w zależności od ilości metrów kwadratowych danego lokalu. W rzeczywistości na cenę mieszkania wpływa dużo więcej czynników, takich jak między innymi odległość od centrum, ilość przystanków, sklepów w okolicy czy możliwość zaparkowania pojazdu. Omawiany przez nas problem skupia się tylko i wyłącznie na wielkości mieszkania, ponieważ uznalismy to jako jeden z głównych czynników, z których wynika cena mieszkania. Oczywiście spadek gradientowy pozwala na dodanie dużo większej ilości danych wejściowych, wręcz nieskończonej (może to spowodować przeuczenie się algorytmu, co będzie powodowało dobre odwzorowanie tylko i wyłącznie dla zbioru testowego, a nie rzeczywistych danych). Zdecydowaliśmy jednak tak, ponieważ dzięki braniu pod uwagę tylko jednej danej wejścowej jesteśmy w stanie przedstawić każdy moment algorytmu używając wykresów w przestrzeni trójwymiarowej co naszym zdaniem jest dużo lepsze dla czytelnika, który pierwszy raz spotyka się z tym algorytmem.

2.2 Ograniczenia algorytmu

Oczywiście jak większość algorytmów i ten posiada ograniczenia. Jak w większości algorytmów związanych z uczeniem maszynowym i sztuczną inteligencją bardzo łatwe jest wystąpienie niedouczenia lub przeuczenia. Pierwszy z problemów wystąpi na pewno - tak jak wspominaliśmy w poprzednim podpunkcie specjalnie nasz algorytm nie będzie idealnie odwzorowywał w pełni rzeczywistości ze względu na małą ilość parametrów wejściowych. Jednakże niedouczenie może wystąpić też z kilku innych powodów - jednym z głównych problemów algorytmów uczenia maszynowego jest zbiór danych, zazwyczaj posiadanie wystarczająco dużego i reprezentatywnego zbioru danych to jest 90% sukcesu. 10% jest to tylko i wyłącznie wybranie odpowiedniej metody oraz odpowiednie zmodyfikowanie jej pod dany problem. Kolejnym problemem jest przeuczenie naszeo algorytmu - z tym można poradzić sobie dużo łatwiej. Jedną z technik jest podzielenie zbioru danych na 3 części - 60% danych są przeznaczane do nauczenia naszego algorytmu, 20% do uzyskania najlepszej regresji liniowej oraz 20% do ostatecznego przetestowania wynikowej. Przy przeuczeniu algorytmu możemy też spróbować zmniejszyć ilość parametrów wejściowych lub zmniejszych wagę części z nich używając tak zwanej regulacji spadku gradientowego.

3 Zbiór danych

3.1 Pochodzenie danych

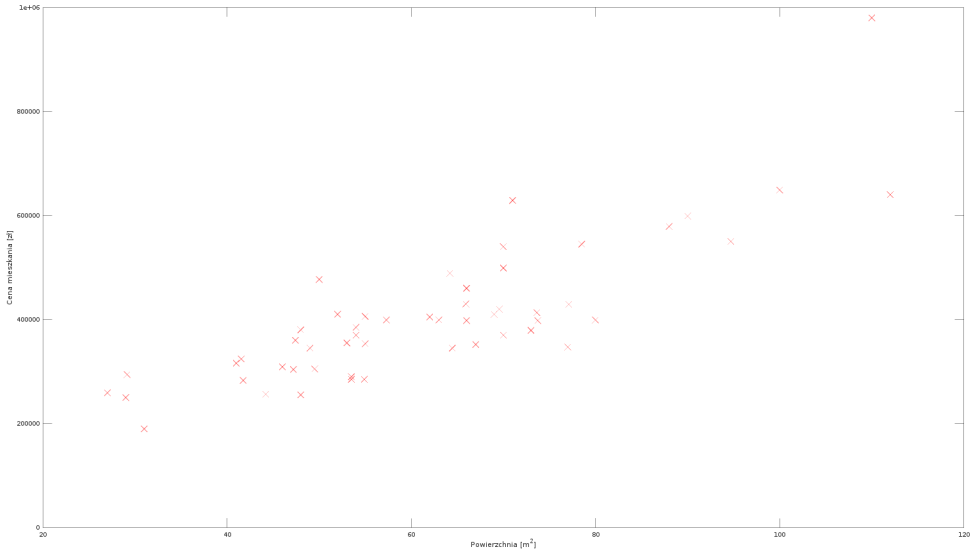
Dane używane przez nas do algorytmu to para liczb - cena mieszkania oraz jego ilość metrów kwadratowych. Wszystkie dane zostały spisane z ogólnodostępnej strony otodom, gdzie każdy może wystawić mieszkanie na sprzedaż. Nie było żadnych innych kryteriów od tego, aby mieszkanie znajdowało się we Wrocławiu - są tu jednocześnie mieszkania używane jak i nowe budownictwo. Link do srony, z której zostały pobrane dane:

`https://www.otodom.pl/sprzedaz/mieszkanie/wroclaw/`

Zebranych zostało 53 przykładów danych, z tego 3 zostaną przeznaczone na odrzucenie najbardziej odbiegających danych - jest to celowy zabieg, który pozwala usunąć około 5% wyników, które znacząco odbiegają od normy oraz mogą spowodować niepoprawne nauczenie algorytmu, co jest równoznaczne ze źle wyznaczoną regresją liniową. Wszystkie dane zostały zebrane losowo, nie były w żaden sposób poddane modyfikacji oraz selekcji wstępnej.

3.2 Zoobrazowanie zbioru danych

Na poniższym wykresie przedstawiono wszystkie dane, które zostały zebrane. Jest to 53 punktów, dla których położenie jest określone poprzez wartość na osi ceny oraz powierzchni mieszkania.



Rysunek 1: Surowy zbiór danych

Wszystkie dane z powyższego wykresu zostały również zebrane w poniższej tabeli. Są one ułozone w sposób nieposortowane - dokładnie w takiej samej kolejności w jakiej zostały znalezione w wyszukiwarce otodom.pl. Przy cenach mieszkań zostały odcięte wartości po przecinku, natomiast wartości powierzchni zostały niezmienione względem danych podanych na stronie - dane te mogę być lekko różne od prawdziwego stanu w danym mieszkaniu.

Cena lokalu [zł]	Powierzchnia $[m^2]$	Cena lokalu [zł]	Powierzchnia $[m^2]$
649 000	100.00	399 000	63.00
550 000	94.70	353 376	55.00
499 000	70.00	355 000	53.00
540 000	70.00	599 000	90.00
309 000	46.00	259 000	27.00
489 000	64.20	410 000	52.00
460 000	66.00	399 000	80.00
351 972	67.00	398 000	73.73
413 200	73.63	285 000	53.50
347 000	77.00	369 900	54.00
255 000	48.00	316 000	41.00
305 000	49.50	250 000	29.00
370 000	70.00	303 932	47.20
324 012	41.54	345 000	49.00
290 000	53.50	284 900	54.90
256 180	44.20	405 900	55.00
404 900	62.00	283 000	41.75
410 000	69.00	429 000	77.10
629 000	71.00	399 000	57.30
640 000	112.00	980 000	110.00
345 000	64.46	189 700	31.00
380 000	48.00	430 000	65.93
420 000	69.55	360 000	47.42
379 000	73.00	385 000	54.00
579 000	88.00	398 000	66.00
294 000	29.15	545 000	78.51
477 000	50.00	-	-

4 Filtracja danych

4.1 Sposób filtracji

Do danych zbieranych w projekcie często dojdą takie, które mogą zaburzyć otrzymanie prawidłowego wyniku. Jedną z metod, aby poradzić sobie z tym jest filtracja danych poprzez odrzucenie tych najbardziej skrajnych. Danych można odrzucić nawet połowę próbek, aczkolwiek przypadek przedstawiony tutaj nie potzrebuje aż tak rygorystycznych reguł. Dane, które odrzucimy to będą 3 próbki osób, które po prostu zawyżyły cenę swoich mieszkań. Ważnym jest, żeby przefiltrować dane w odpowiedni sposób odrzucając te próbki, które mogą mieć najbardziej negatywny wpływ na wynik końcowy.

4.2 Odrzucenie danych

Skoro celem naszego algorytmu jest znalezienie zależności między ceną mieszkania, a jego metrażem, możemy stwierdzić w uproszczeniu, że poszukujemy tak naprawdę średniej ceny za metr kwadratowy mieszkania we Wrocławiu.

y	cena mieszkania$[zl]$
X	powierzchnia mieszkania$[m^2]$
$z = \frac{y}{X}$	cena za metr kwadratowy$[\frac{zl}{m^2}]$

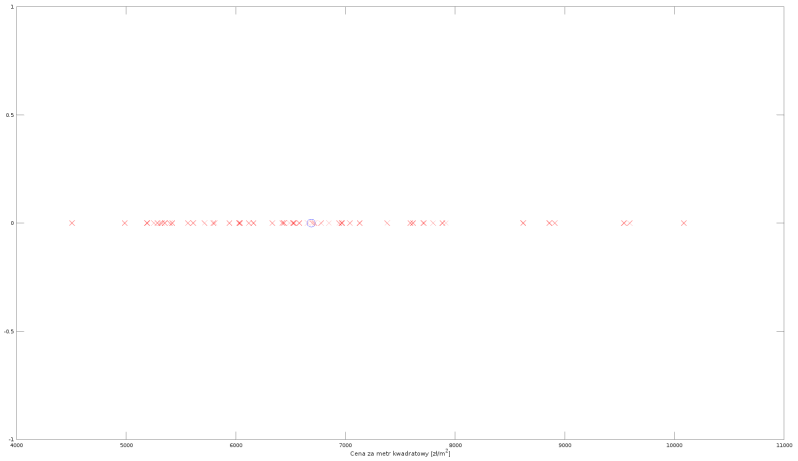
Najbardziej skrajne dane obliczamy za pomocą sumowania wszystkich 'z' oraz podzielenia tego przez liczbę próbek. W ten sposób znajdujemy średnią wartość kosztu za metr kwadratowy. Mozna sądzić, że w tym momencie mamy rozwiązany problem regresji liniowej, bo wystarczy skalować to na prostą. Po części tak, ale problemem tutaj nie jest samo znalezienie prostej, a opisanie jednej z metod pozyskiwania prostej opisującej jakieś dane.

s	średnia cena za $m^2[zl]$
$s = 6315,9$	
d_i	odchylenie od średniej $[zl]$
$d_i = \lvert s - z_i \rvert$	

W ten sposób otrzymamy d_i dla każdego elementu. Poszukujemy tych, których odchył jest największy. Według obliczeń są to odpowiednio pary:

- 29.15 metrów za 294 000 zł
- 27 metrów za 259 000 zł
- 50 metrów za 477 000 zł

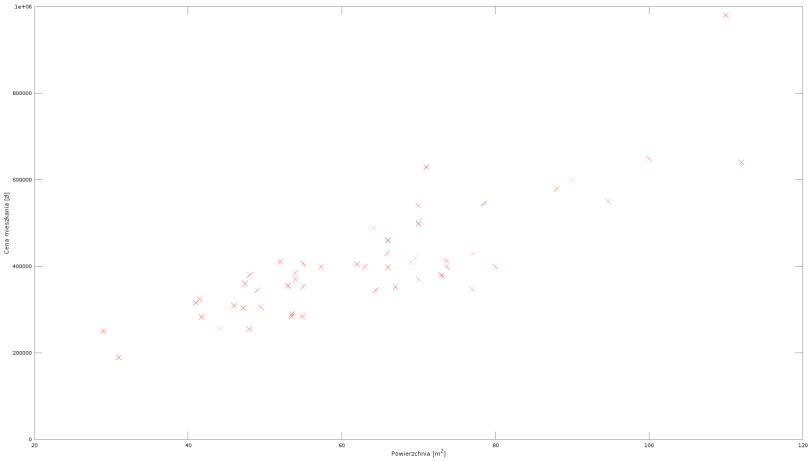
Dokładne zoobrazowanie tych danych jest przedstawione na poniższym wykresie 1D. 3 punkty stanowczo są odsunięte od średniej ceny za metr kwadratowej (niebieskie kółko).



Rysunek 2: Wykres ceny metra kwadratowego dla wszystkich próbek

4.3 Dane po filtracji

Jak widać filtracja odrzuciła nam trzy punkty maksymalnie po prawo. Nasz wykres został zmodyfikowany oraz te 3 próbki zostały usunięte z próbek, które będą używane podczas spadku gradientowego.



Rysunek 3: Wykres ceny metra kwadratowego dla wszystkich próbek

Po zobaczeniu nowego wykresu można zastanowiś się - czemu trzy punkty, które były skrajne i najbardziej oddalone od pozostałych nie zostały usunięte? To wszystko z powodu tego jaka była nasza filtracja - nie skupialiśmy się na punktach, które leżą najdalej od głównego skupiska punktów, a te których cena za metr kwadratowy jest najbardziej oddalona od średniej. Na przykładzie:
Posiadamy cztery punkty i jeden z nich musimy odrzucić:

- 100 metrów za 500 000 zł
- 60 metrów za 240 000 zł
- 50 metrów za 275 000 zł
- 40 metrów za 220 000 zł

$z_1 = 500000/100 = 5000[zl]$
$z_2 = 240000/60 = 4000[zl]$
$z_3 = 275000/50 = 5500[zl]$
$z_4 = 220000/40 = 5500[zl]$
$s = (5000 + 4000 + 5500 + 5500)/4 = 5000[zl]$
$d_1 = 0[zl]$
$d_2 = 1000[zl]$
$d_3 = 500[zl]$
$d_4 = 500[zl]$

Jak widać na powyższym przykładzie odrzucona będzie wartość nie ta, która leży najdalej od reszty, gdybyśmy przedstawili ją na wykresie 2D, a ta której cena za metr kwadratowy najbardziej odbiega od średniej cen. Ważne jest by średnią cen za metr kwadratowych sporządzić na przykładzie cen z każdej próbki, a nie zsumować cenę wszystkich metrów kwadratowych oraz cen mieszkań - dzięki temu unikamy podwyższania ważności próbek, które posiadają wiekszy metraż.
Dzięki filtracji pozbywamy się próbek, które mogą zaszkodzić naszemu algorytmowi. Zawsze warto zrobić filtrację danych, pod warunkiem, że będzie ona przemyślana i na pewno poprawna - dla przykładu trudno zrobić filtrację przy rozpoznawaniu cyfr, dlatego czasami lepiej warto ją ominąć i skupic się na samym algorytmie.

5 Funkcja kosztu

Według definicji funkca kosztów jest to funkcja, którą za pomocą algorytmu chcemy zminimalizować. Często jest reprezentowana jako róznica między wartością wyjściową algorytmu, a wartością rzeczywistą. Generalna idea mówi, że funkcja kosztu zwraca dystans od prawdziwej wartości, a algorytm stara się by dystans był tam mały, aby był uznawany jako nieistotny.
Definicja jest trochę pogmatwana - dla naszego przypadku funkcja kosztu będzie wskaźnikiem jak dobry jest nasz algorytm. Skoro ma on na celu najlepsze przewidywania cen mieszkań w takim razie funkcja kosztu będzie najlepsza, kiedy nasza regresja liniowa będzie najlepiej obrazować aktualne oraz przyszłe dane.
Dla regresji liniowej funkcja kosztu jest to kwadrat błędu między wartością wyjściową, a wartością rzeczywistą. Dużo łatwiej jest to zoobrazować wzorem:

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} (\hat{y_i} - y_i)^2$$

Gdzie:

m	ilość próbek
$\hat{y_i}$	przewidywana cena
y_i	rzeczywista cena

Wróćmy do naszych przykładowych danych z punktu **4.3**. Obliczymy dwie funkcje kosztu, dla:

- $h_1(x_i) = 275000$
- $h_2(x_i) = \theta_0 + \theta_1 * x_i$ dla $\theta = [-50000\ 4500]$

Przypomnijmy dane, po filtracji zostały nam tylko trzy punkty:

- 100 metrów za 500 000 zł
- 50 metrów za 275 000 zł
- 40 metrów za 220 000 zł

Dla pierwszej predykcji obliczenia są banalne, ponieważ nasza prezwidywana prosta jest linią równoległą do osi X. W takim razie nasza funkcja kosztu to:

$$J(275000, 0) = \frac{1}{6} * ((275000 - 500000)^2 + (275000 - 275000)^2 + (275000 - 220000)^2)$$

Co dajwe wynik $J = 29108333333$ - jak widać jest to bardzo duża liczba, co świadczy o tym, że nie jest to najlepsza regresja liniowa. Dla drugie przypadku jest trochę więcej liczenia, dlatego pierw obliczymy predykcję, a następnie podstawimy do całego wzoru.

$h(x_1) = -50000 + 4500 * 100 = 500000$
$h(x_2) = -50000 + 4500 * 50 = 275000$
$h(x_3) = -50000 + 4500 * 40 = 230000$

Jak widać predykcja dla dwóch wartości jest idealna, zobaczmy w takim razie ile będzie wynosić funkcja kosztu:

$$J(275000, 0) = \frac{1}{6} * ((500000 - 500000)^2 + (275000 - 275000)^2 + (230000 - 220000)^2)$$

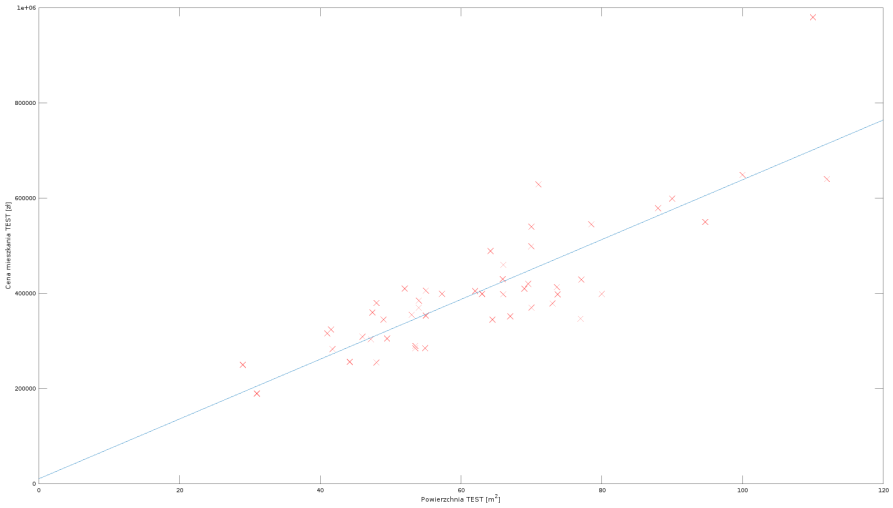
Wynik $J = 100000000$ również jest spory, ale dużo mniejszy niż dla pierwszej funkcji. Wielkości wyników spowodowane są tym, że operujemy na danych, które są w okolicach ćwierć do pół miliona. Jak w takim razie zadecydować, która predykcja jest lepsza? Ta, która jest mniejsza, ponieważ aktualnie jest ona bliższa osiągnięcia minimum. W taki sam sposób algorytm wybiera lepszą funkcję kosztu, aczkolwiek nie musimy wprowadzić kilkudziesięciu, kilkuset wzorów na predykcję, a spadek gradientowy sam znajduje optymalne rozwiązanie - ale o tym dokładniej z następnym punkcie.

6 Spadek gradientowy

Spadek gradientowy jest to jeden ze sposobów odnalezienia najlepszej regresji. Używa on do tego funkcję kosztu, do której parametry modyfikuje sobie w zależności od kroku (α) oraz pochodnej po kierunku. Wzór na spadek gradientowy powtarzamy aż do otrzymania wszystkich pochodnych równych 0 lub określoną ilość iteracji.

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$$

W zależności od ilości θ będziemy musieli równocześnie aktualizować tyle nowych θ. Wzór jest w pewnym sensie uproszczeniem, tak naprawdę po lewej stronie nie będą stały θ_j, a $\theta_j temp$, a na samym końcu wszystkie θ będą uaktualniane.



Rysunek 4: Idealna regresje znaleziona przez spadek gradientowy

Powyższy wykres pokazuje znalezioną funkcje kosztu - nie istnieje inna funkcja kosztów z innymi parametrami, która lepiej opisałaby te dane. W takim razie można powiedziec, że nie zostało tylko osiągnięte lokalne ekstremum, a nawet globalne minimum.

Jak w takim razie wygląda dokładnie algorytm spadku gradientowego? Po pierwsze nasz algorytm trwa określoną ilość iteracji. W pierwszym momencie można pomyśleć, że kończy się zbyt wcześnie i nie dochodzi do minium, aczkolwiek praca nad nim polega właśnie na dopasowaniu parametrów - kroku spadku oraz ilości iteracji.

```
for iteracja = 1 : ilosc_iteracji
   h = X * theta
   theta = theta - alpha * (1/m) * (X' * (h - y))
end
```

Jak widać mimo całej złozoności problemu sam algorytm jest bardzo prosty. Wcześniej wspominaliśmy o tym, że trzeba sprawdzić czy algorytm na pewno znajduje minimum. Można to zrobić rysując zależność funkcji kosztu od poszczególnych theta. Nie jesteśmy w stanie narysować pełnej funkcji, ale dzięki zachowaniu parametrów takich jak historyczne dane J oraz θ. Jeżeli widać, że dla naszego wykresu w pewnym momencie J jest stałe oznacza to, że doszliśmy do momentu, gdy osiągneliśmy szukane przez nas θ_0 oraz θ_1.

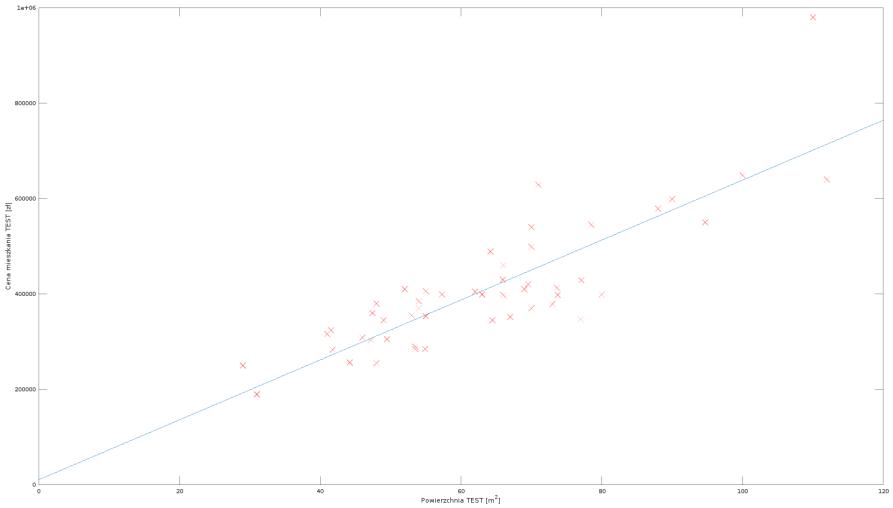
Kied w takim razie kończy się spadek gradientowy? Dzięki matematyce wiemy, że minimum jest to miejsce, w którym osiągamy najniżej położony punkt, czyli przed i po w każdym kierunku powinniśmy posiadać wartości większe. Dzięki pochodnej wiemy czy to jest ten punkt, ponieważ pochodne po każdym kierunku powinny wynosić 0.

7 Podsumowanie

Problem regresji liniowej za pomocą spadku gradientowego został poprawnie rozwiązany. Wszystkie kroki zostały przeprowadzone pomyślnie oraz otrzymany prostą opisującą próbki tak, aby dzięki niej mozna było przewidzieć kolejne wartości:

$$h(x) = 10473 + 6282.5 * x$$

Pewność tego wyniku potwierdza utknięcie $J(\theta)$ w globalnym minimum. Jednocześnie uzyskany wykres już bez dokładnego przyglądania się obrazuje to, że uzyskana prosta jak najbardziej opisuje te próbki i przewidzi z małym błędem dodane w przyszłości.



Rysunek 5: Idealna regresje znaleziona przez spadek gradientowy

Na sam koniec podsumowujmy, dlaczego spaek gradientowy jest jedną z lepszych metod znajdywania predykcji. Po pierwsze możemy posiadać nieskończoną ilość danych wejściowych - nie tylko ze względu na ilość, ale też ich rodzaj (np. metraż, odległość od centrum). Po drugie jesteśmy w stanie łatwo sprawdzić czy nasz algorytm zadziałał poprawnie. Jesteśmy również w stanie w łatwy sposób przeprowadzić badania na nowych próbkach, ponieważ posiadamy rzeczywiste odwzorowanie i nie musimy uruchamiać algorytmu przy każdej nowej próbce, a wystarczy użyć uzyskanego wzoru.